BECKHOFF New Automation Technology

Documentation | EN

BC8050 and BC8150

Bus Terminal Controller with RS485 or RS232 Interface

Table of contents

BECKHOFF

1   Foreword

1.1   Notes on the documentation

Intended audience

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning these components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.

We reserve the right to revise and change the documentation at any time and without prior announcement.

No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH. Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents: EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702 with corresponding applications or registrations in various other countries.

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.
The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization are prohibited.
Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design.

1.2 Safety instructions

Safety regulations

Please note the following safety instructions and explanations!
Product-specific safety instructions can be found on following pages or in the areas mounting, wiring, commissioning etc.

Exclusion of liability

All the components are supplied in particular hardware and software configurations appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

Personnel qualification

This description is only intended for trained specialists in control, automation and drive engineering who are familiar with the applicable national standards.

Description of instructions

In this documentation the following instructions are used.
These instructions must be read carefully and followed without fail!

⚠ DANGER
Serious risk of injury!
Failure to follow this safety instruction directly endangers the life and health of persons.

⚠ WARNING
Risk of injury!
Failure to follow this safety instruction endangers the life and health of persons.

⚠ CAUTION
Personal injuries!
Failure to follow this safety instruction can lead to injuries to persons.

NOTE
Damage to environment/equipment or data loss
Failure to follow this instruction can lead to environmental damage, equipment damage or data loss.

Tip or pointer

This symbol indicates information that contributes to better understanding.

1.3   Documentation issue status

Version	Comment
3.2.0	• Technical data updated
	• Ex marking added to technical data
	• Chapter *ATEX - Special conditions (extended temperature range)* added
	• Chapter *Instructions for ESD protection* added
	• Chapter *Disposal* added
	• New title page
3.1.0	• Technical data updated
	• ATEX added
3.0.0	• Migration
2.0.0	• BC8050 added
1.1.1	• ADS communication example added [▶ 86]
1.1.0	• Notes to meet the UL requirements added.
1.0.1	• Minor routine corrections
1.0.0	• First public issue (only available in German)

BC8150 firmware

For updating your firmware you need a serial cable, the KS2000 configuration software, or the firmware update program.

Document version	BC8050		BC8150	
	Firmware version	Hardware version	Firmware version	Hardware version
3.2.0	B1	09	B4	15
3.1.0	B0	05	B1	09
3.0.0	B0	04	B1	08
2.0.0	B0	00	B1	04
1.1.1	-	-	B0	00
1.1.0	-	-	B0	00
1.0.1	-	-	B0	00
1.0.0	-	-	B0	00

The firmware and hardware versions (delivery state) can be found on the sticker on the underside of the Bus Terminal Controller.

2 Product overview

2.1 Bus Terminal Controllers of the BCxx50 series

Bus Terminal Controllers are Bus Couplers with integrated PLC functionality. The BCxx50 Bus Terminal Controllers have a fieldbus interface, are intelligent slaves and can be used as decentralized intelligence within the system. They are located in a cost-optimized and compact housing. In contrast to the BCxx00 range, the BCxx50 range supports up to 255 Bus Terminals via the K-Bus extension.

The Bus Terminal Controller is programmed using the TwinCAT programming system according to IEC 61131-3. The BCxx50 configuration/programming interface is used for loading the PLC program. If the TwinCAT software PLC is in use, the PLC program can also be loaded via the fieldbus.

The inputs and outputs of the connected Bus Terminals are assigned in the default setting of the mini-PLC. Each individual Bus Terminal can be configured in such a way that it exchanges data directly through the fieldbus with the higher-level automation device. Similarly, pre-processed data can be exchanged between the Bus Terminal Controller and the higher-level controller via the fieldbus.

Fieldbus interface

The variants of the BCxx50 series Bus Terminal Controllers differ in terms of their fieldbus interfaces. Various versions cover the most important fieldbus systems:

- BC3150: PROFIBUS DP
- BC5150: CANopen
- BC5250: DeviceNet
- BC8050: RS485, various protocols
- BC8150: RS232, various protocols

Programming

The BCxx50 devices are programmed according to the powerful IEC 61131-3 standard. Like for all other BECKHOFF controllers, the TwinCAT automation software is the basis for parameterization and programming. Users therefore have the familiar TwinCAT tools available, e.g. PLC programming interface, System Manager and TwinCAT Scope. Data is exchanged optionally via the serial interface (COM1) or via the fieldbus through Beckhoff PC FCxxxx fieldbus cards.

Configuration

The configuration is also carried out using TwinCAT. The fieldbus interface can be configured and parameterized via the System Manager. The System Manager can read all connected devices and Bus Terminals. After the parameterization, the configuration is saved on the BCxx50 via the serial interface. The configuration thus created can be accessed again later.

2.2   BC8050 - Introduction

Fig. 1: BC8050

The BC8050 Bus Terminal Controller with serial RS485 interface extends the Beckhoff small controller series by a cost-optimized version in a compact housing. An open serial protocol – like in the BK8x00 Bus Couplers – and the Modbus RTU/ASCII protocol are implemented. The address and the protocol are selected via the two rotary selection switches.

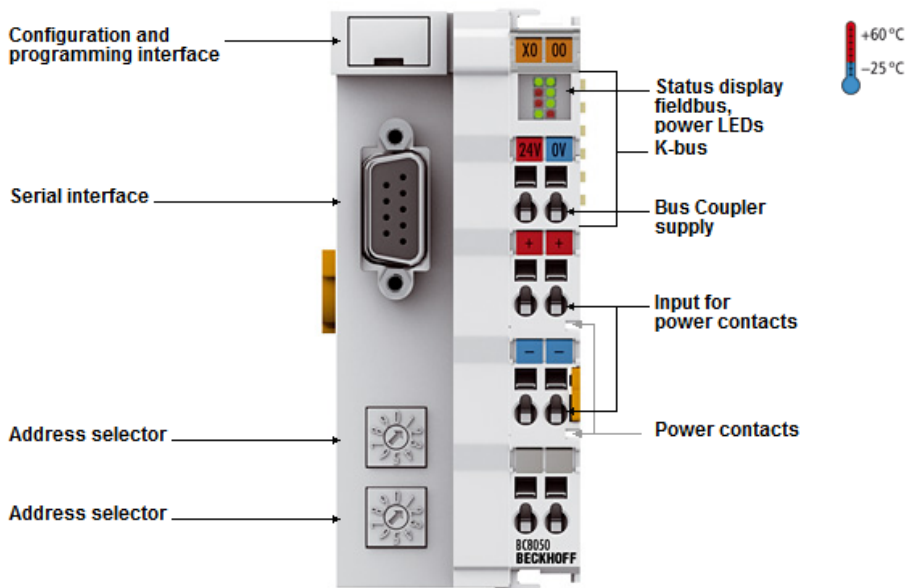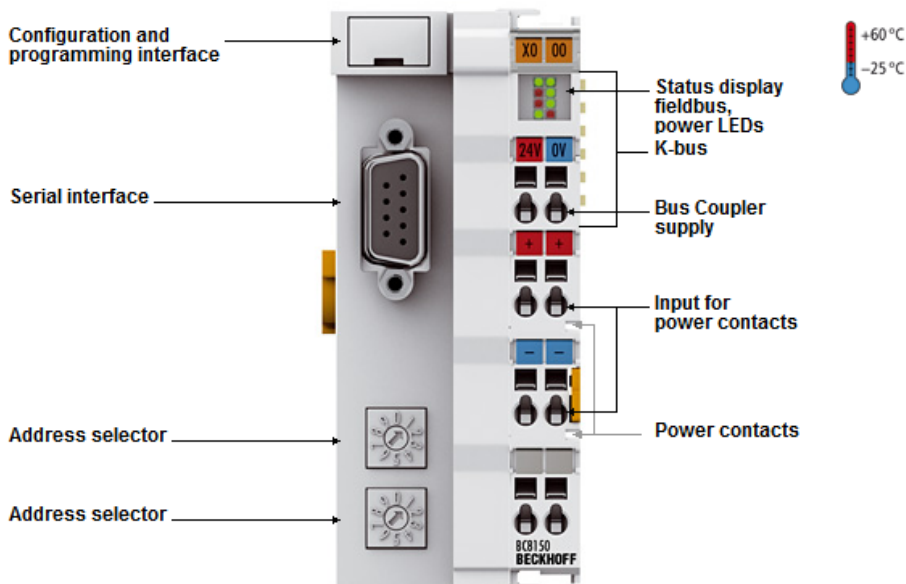## 2.3   BC8150 - Introduction

Fig. 2: BC8150

The BC8150 Bus Terminal Controller with serial RS232 interface extends the Beckhoff small controller series by a cost-optimized version in a compact housing. An open serial protocol – like in the BK8x00 Bus Couplers – and the Modbus RTU/ASCII protocol are implemented. The address and the protocol are selected via the two rotary selection switches.

2.4 Technical data

2.4.1 Technical data - BC8x50

Technical data	BC8050	BC8150
Processor	16 bit micro-controller	
Diagnostics LEDs	2 x power supply, 2 x K-Bus	
Configuration and programming software	TwinCAT PLC	
Fieldbus	RS485: - KS8000 protocol - ModbusRTU - ModbusASCII	RS232 - Serial ADS - KS8000 protocol - ModbusRTU - ModbusASCII
Serial interface	COM1 (RS232 for configuration and programming, automatic baud rate detection 9600/19200/38400 baud)	
Number of bus terminals	64 (255 with K-bus extension)	
Digital peripheral signals	2040 inputs/outputs	
Analog peripheral signals	1024 inputs/outputs	
Configuration possibility	via TwinCAT or the controller	
Maximum fieldbus byte number	depending on fieldbus	
Maximum number of bytes - PLC	2048 bytes of input data, 2048 bytes of output data	
Bus connection	D-sub, 9-pin	

Technical data	BC8050	BC8150
Power supply (Us)	24 V_{DC} (-15 % /+20 %)	
Input current (Us)	60 mA + (total K-bus current)/4	
Starting current	approx. 2.5 x continuous current	
K-bus current (5 V)	maximum 1750 mA	
Power contact voltage (Up)	maximum 24 V_{DC}	
Power contact current load (Up)	maximum 10 A	
Recommended back-up fuse (Up)	≤10 A	
Dielectric strength	500 V (power contact/supply voltage/fieldbus)	

⚠ **CAUTION**
UL-Requirements For power supply of the Bus Terminal Controller (Us) use a 4 A fuse or an *NEC Class 2* power supply to meet the UL requirements!

Technical data	BC8050	BC8150
Weight	approx. 100 g	
Dimensions (W x H x D)	approx. 44 mm x 100 mm x 68 mm	
Permissible ambient temperature range during operation	-25 °C ... +60 °C	
Permissible ambient temperature range during storage	-40°C ... +85°C	
Relative humidity	95 % no condensation	
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27	
EMC immunity / emission	conforms to EN 61000-6-2 / EN 61000-6-4	
Installation position	variable	
Protection class	IP20	
Approvals/markings*	CE, UKCA, cULus, EAC, ATEX [▶ 24]	

*) Real applicable approvals/markings see type plate on the side (product marking).

Ex marking

Standard	Marking
ATEX	II 3 G Ex nA IIC T4 Gc

2.4.2   Technical data - RS485

System data	RS485 (BC8050)
Protocols	• Address 1-39 serial ADS*, KS8000 protocol (automatic recognition of the protocol and the baud rate (9600, 19200, 38400 baud, fixed data frame, 8 data bits, even parity, one stop bit) • Address 40-59 ModbusRTU (fixed baud rate 1200, 2400, 4800, **9600**, 19200, 38400, DEFAULT 9600 baud, 8 bit data, optionally even, odd, **no** default no parity)
Number of I/O modules	32 (99 with repeater)
Number of I/O points	depending on controller
Data transfer medium	shielded copper cable 2 x 0.25 mm², cable type A according to EN 50170
Segment length	up to 1200 m
Data transfer rate	1200 - 38400 baud
Topology	RS485 line
Transmission time	depending on protocol and baud rate

*) only via the configuration interfaces, not via RS485.

2.4.3   Technical data - RS232

System data	RS232 (BC8150)
Protocols	• Address 1-39 serial ADS, KS8000 protocol (automatic recognition of the protocol and the baud rate (9600, 19200, 38400 baud, fixed data frame, 8 data bits, even parity, one stop bit) • Address 40-59 ModbusRTU (fixed baud rate 1200, 2400, 4800, **9600**, 19200, 38400, DEFAULT 9600 baud, 8 bit data, optionally even, odd, **no** default no parity)
Number of I/O modules	1
Number of I/O points	depending on controller
Data transfer medium	shielded copper cable, 3 x 0.25 mm²
Segment length	up to 15 m
Data transfer rate	1200 - 38400 baud
Topology	RS232 peer to peer
Transmission time	depending on protocol and baud rate

2.4.4   Technical data - PLC

PLC data	BCxx5x
Programmability	via serial programming interface or via the fieldbus
Program memory	48 kbyte
Source code memory	128 kbyte
Data memory	32 kbyte
Remanent flags	2 kbyte
PLC cycle time	Approx. 3.0 ms for 1000 IL commands (without I/O cycle)
Programming languages	IEC 6-1131-3 (IL, LD, FBD, ST, SFC)
Runtime	1 SPS Task
Online Change	Yes
Up/Down Load Code	Yes/Yes

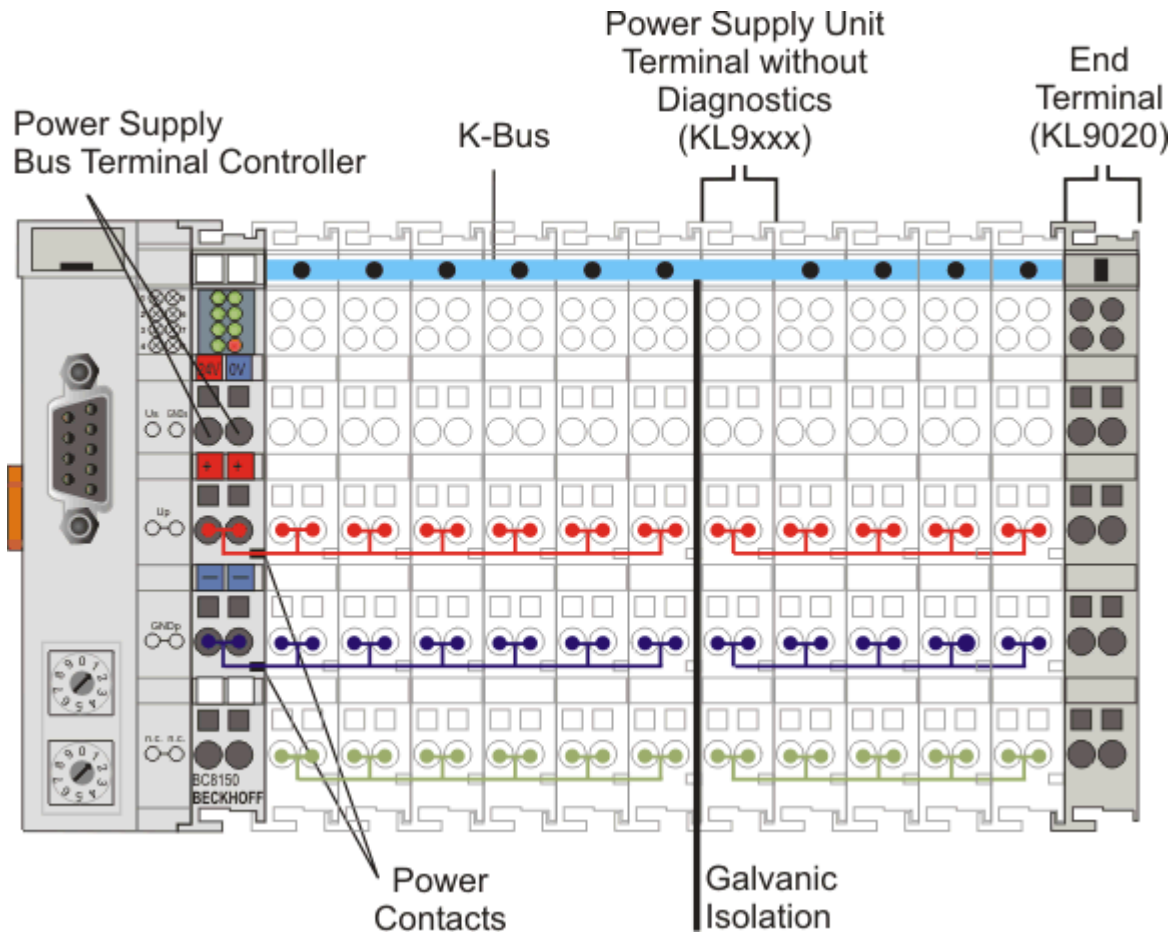## 2.5   The principle of the Bus Terminal

Fig. 3: The principle of the Bus Terminal

2.6   The Beckhoff Bus Terminal system

Up to 256 Bus Terminals, with 1 to 16 I/O channels per signal form

The Bus Terminal system is the universal interface between a fieldbus system and the sensor / actuator level. A unit consists of a Bus Coupler as the head station, and up to 64 electronic series terminals, the last one being an end terminal. Up to 255 Bus Terminals can be connected via the K-Bus extension. For each technical signal form, terminals are available with one, two, four or eight I/O channels, which can be mixed as required. All the terminal types have the same mechanical construction, so that difficulties of planning and design are minimized. The height and depth match the dimensions of compact terminal boxes.

Decentralized wiring of each I/O level

Fieldbus technology allows more compact forms of controller to be used. The I/O level does not have to be brought to the controller. The sensors and actuators can be wired decentrally, using minimum cable lengths. The controller can be installed at any location within the plant.

Industrial PCs as controllers

The use of an Industrial PC as the controller means that the operating and observing element can be implemented in the controller's hardware. The controller can therefore be located at an operating panel, in a control room, or at some similar place. The Bus Terminals form the decentralized input/output level of the controller in the control cabinet and the subsidiary terminal boxes. The power sector of the plant is also controlled over the bus system in addition to the sensor/actuator level. The Bus Terminal replaces the conventional series terminal as the wiring level in the control cabinet. The control cabinet can have smaller dimensions.

Bus Couplers for all usual bus systems

The Beckhoff Bus Terminal system unites the advantages of a bus system with the possibilities of the compact series terminal. Bus Terminals can be driven within all the usual bus systems, thus reducing the controller parts count. The Bus Terminals then behave like conventional connections for that bus system. All the performance features of the particular bus system are supported.

Mounting on standardized mounting rails

The installation is standardized thanks to the simple and space-saving mounting on a standardized mounting rail (EN 60715, 35 mm) and the direct wiring of actuators and sensors, without cross connections between the terminals. The consistent labelling scheme also contributes.

The small physical size and the great flexibility of the Bus Terminal system allow it to be used wherever a series terminal is also used. Every type of connection, such as analog, digital, serial or the direct connection of sensors can be implemented.

Modularity

The modular assembly of the terminal strip with Bus Terminals of various functions limits the number of unused channels to a maximum of one per function. The presence of two channels in one terminal is the optimum compromise of unused channels and the cost of each channel. The possibility of electrical isolation through potential feed terminals also helps to keep the number of unused channels low.

Display of the channel state

The integrated LEDs show the state of the channel at a location close to the sensors and actuators.

K-Bus

The K-Bus is the data path within a terminal strip. The K-Bus is led through from the Bus Coupler through all the terminals via six contacts on the terminals' side walls. The end terminal terminates the K-Bus. The user does not have to learn anything about the function of the K-Bus or about the internal workings of the terminals and the Bus Coupler. Many software tools that can be supplied make project planning, configuration and operation easy.

Potential feed terminals for isolated groups

The operating voltage is passed on to following terminals via three power contacts. You can divide the terminal strip into arbitrary isolated groups by means of potential feed terminals. The potential feed terminals play no part in the control of the terminals, and can be inserted at any locations within the terminal strip.

Up to 64 Bus Terminals can be used in a terminal block, with optional K-Bus extension for up to 256 Bus Terminals. This count does include potential feed terminals, but not the end terminal.

Bus Couplers for various fieldbus systems

Various Bus Couplers can be used to couple the electronic terminal strip quickly and easily to different fieldbus systems. It is also possible to convert to another fieldbus system at a later time. The Bus Coupler performs all the monitoring and control tasks that are necessary for operation of the connected Bus Terminals. The operation and configuration of the Bus Terminals is carried out exclusively by the Bus Coupler. Nevertheless, the parameters that have been set are stored in each Bus Terminal, and are retained in the event of voltage drop-out. Fieldbus, K-Bus and I/O level are electrically isolated.

If the exchange of data over the fieldbus is prone to errors or fails for a period of time, register contents (such as counter states) are retained, digital outputs are cleared, and analog outputs take a value that can be configured for each output when commissioning. The default setting for analog outputs is 0 V or 0 mA. Digital outputs return in the inactive state. The timeout periods for the Bus Couplers correspond to the usual settings for the fieldbus system. When converting to a different bus system it is necessary to bear in mind the need to change the timeout periods if the bus cycle time is longer.

The interfaces

A Bus Coupler has six different methods of connection. These interfaces are designed as plug connectors and as spring-loaded terminals.

3 Mounting and wiring

3.1 Mounting

⚠ WARNING
Risk of injury through electric shock and damage to the device!
Bring the Bus Terminals system into a safe, de-energized state before starting mounting, disassembly or wiring of the components!

3.1.1 Dimensions

The Beckhoff Bus Terminal system is characterized by low physical volume and high modularity. When planning a project it must be assumed that at least one Bus Coupler and a number of Bus Terminals will be used. The mechanical dimensions of the Bus Couplers are independent of the fieldbus system.
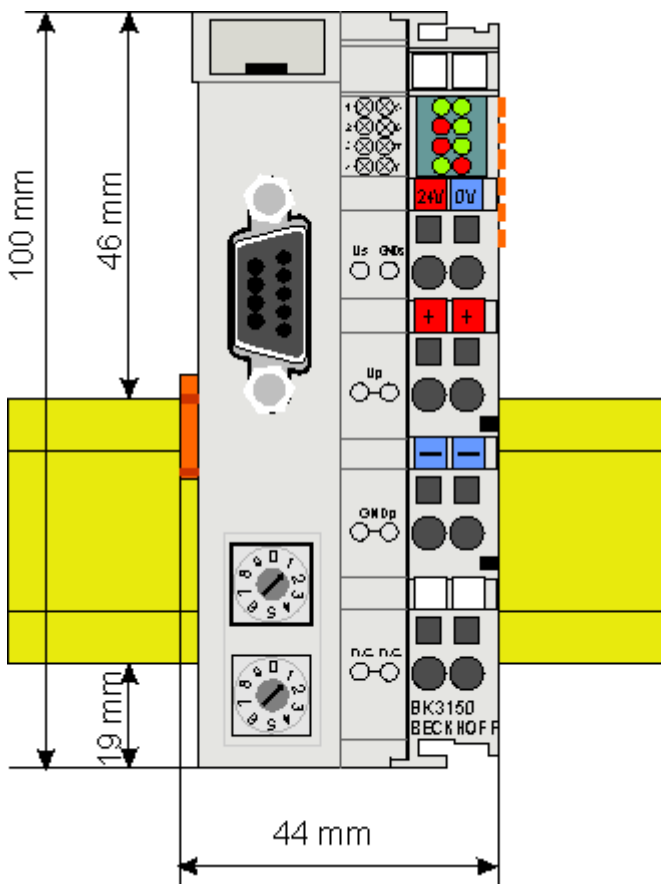

Fig. 4: BCxx50 - Dimensions

The total width of the fieldbus station is the width of the Bus Coupler/Bus Terminal Controller plus the width of the Bus Terminals being used (incl. KL9010 bus end terminal). Depending on design, the Bus Terminals are 12 mm or 24 mm wide. The height is 100 mm.

The BCxx50 series Bus Terminal Controllers are 68 mm deep.

3.1.2   Installation on mounting rails

The Bus Coupler and all the Bus Terminals can be clipped, with a light press, onto a 35 mm mounting rail. A locking mechanism prevents the individual housings from being pulled off again. For removal from the mounting rail the orange colored tension strap releases the latching mechanism, allowing the housing to be pulled off the rail without any force.
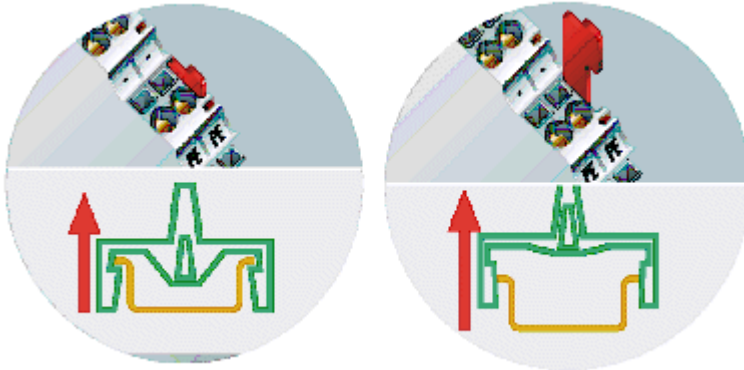
Fig. 5: Release the locking mechanism by pulling the orange tab

Up to 64 Bus Terminals can be attached to the Bus Coupler on the right hand side. When plugging the components together, be sure to assemble the housings with groove and tongue against each other. A properly working connection cannot be made by pushing the housings together on the mounting rail. When correctly assembled, no significant gap can be seen between the attached housings.

Fig. 6: Groove and tongue of the housings

NOTE
Bus Terminals should only be pulled or plugged in switched-off state.
Insertion and removal of Bus Terminals is only permitted when switched off. The electronics in the Bus Terminals and in the Bus Coupler are protected to a large measure against damage, but incorrect function and damage cannot be ruled out if they are plugged in under power.

3.2 Wiring

> **⚠ WARNING**
>
> **Risk of injury through electric shock and damage to the device!**
>
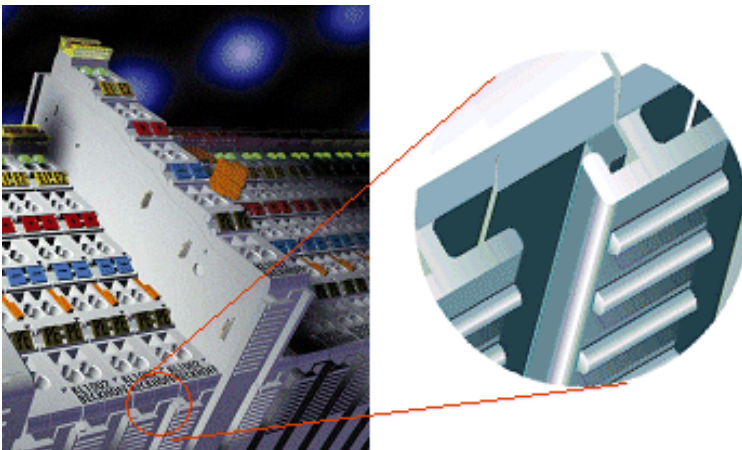> Bring the Bus Terminals system into a safe, de-energized state before starting mounting, disassembly or wiring of the components!

3.2.1 Potential groups, insulation testing and PE

Potential groups

A Beckhoff Bus Terminal block usually has three different potential groups:

- The fieldbus interface is electrically isolated (except for individual Low Cost couplers) and forms the first potential group.
- Bus Coupler / Bus Terminal Controller logic, K-bus and terminal logic form a second electrically isolated potential group.
- The inputs and outputs are supplied via the power contacts and form further potential groups.

Groups of I/O terminals can be consolidated to further potential groups via potential supply terminals or separation terminals.
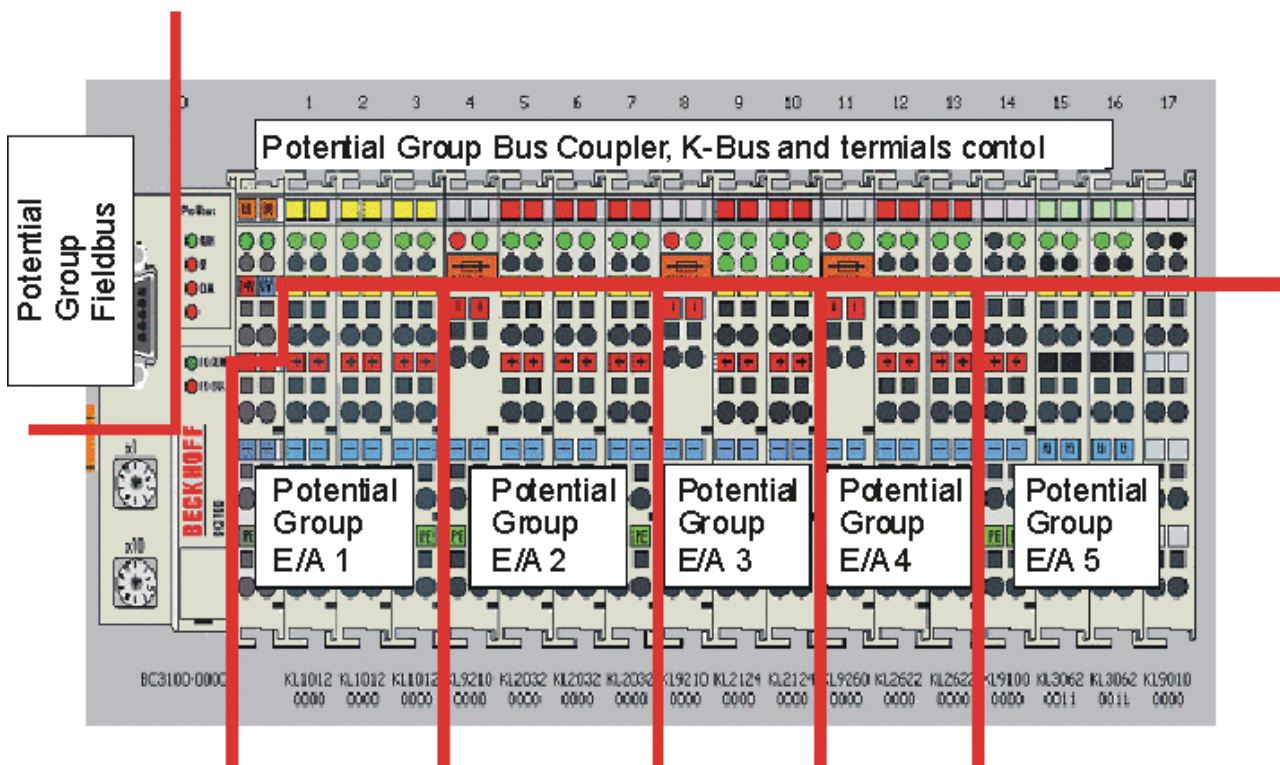
Fig. 7: Potential groups of a Bus Terminal block

Insulation testing

The connection between Bus Coupler / Bus Terminal Controller and Bus Terminals is realized automatically by latching the components. The transfer of the data and the supply voltage for the intelligent electronics in the Bus Terminals is performed by the K-bus. The supply of the field electronics is performed through the power contacts. Plugging together the power contacts creates a supply rail. Since some Bus Terminals (e.g. analog Bus Terminals or 4-channel digital Bus Terminals) are not looped through these power contacts or not completely the Bus Terminal contact assignments must be considered.

The potential feed terminals interrupt the power contacts, and represent the start of a new supply rail. The Bus Coupler / Bus Terminal Controller can also be used for supplying the power contacts.

PE power contacts

The power contact labelled PE can be used as a protective earth. For safety reasons this contact mates first when plugging together, and can ground short-circuit currents of up to 125 A.

Fig. 8: Power contact on the left

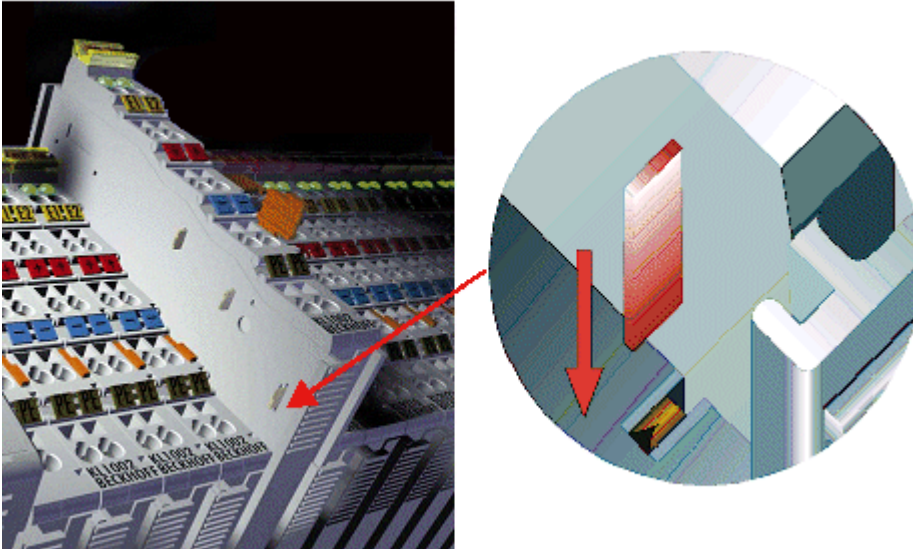It should be noted that, for reasons of electromagnetic compatibility, the PE contacts are capacitively coupled to the mounting rail. This can both lead to misleading results and to damaging the terminal during insulation testing (e.g. breakdown of the insulation from a 230 V power consuming device to the PE conductor). The PE supply line at the Bus Coupler / Bus Terminal Controller must be disconnected for an insulation test. In order to uncouple further feed locations for the purposes of testing, the feed terminals can be pulled at least 10 mm out from the connected group of other terminals. In that case, the PE conductors do not have to be disconnected.

The power contact with the label PE must not be used for other potentials.

3.2.2   Power supply

⚠ **WARNING**
Risk of injury through electric shock and damage to the device!
Bring the Bus Terminals system into a safe, de-energized state before starting mounting, disassembly or wiring of the components!

Supply of Bus Terminal Controller and Bus Terminals (Us)

The Bus Terminal Controller requires a supply voltage of 24 V_{DC}.

The connection is made by means of the upper spring-loaded terminals labelled *24 V* and *0 V*. This supply voltage is used for the electronic components of the Bus Coupler / Bus Terminal Controllers and (via the K-bus) the electronic components of the Bus Terminals. It is galvanically separated from the field level voltage.
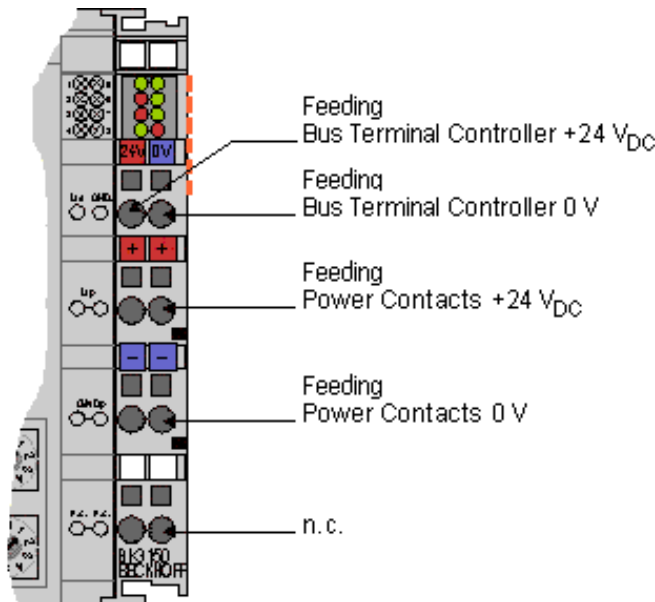
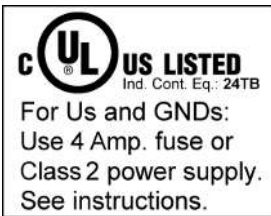Fig. 9: Terminal points for the Bus Terminal Controller supply

Fig. 10: UL identification

⚠ **CAUTION**
Note the UL requirements for the power supply. To comply with the UL requirements, the 24 V_{DC} supply voltage for Us must originate • from an isolated source protected by a fuse of max. 4 A (according to UL248) or • from a voltage supply complying with *NEC class 2*. An *NEC class 2* voltage source must not be connected in series or parallel with another *NEC class 2* corresponding voltage source!

⚠ **CAUTION**
No unlimited voltage sources! To comply with the UL requirements, Us must not be connected with unlimited voltage sources.

Power contacts supply (Up)

The bottom six connections with spring-loaded terminals can be used to feed the supply for the peripherals. The spring-loaded terminals are joined in pairs to a power contact. The feed for the power contacts has no connection to the voltage supply for the BC electronics.

The spring-loaded terminals are designed for wires with cross-sections between 0.08 mm^2 and 2,5 mm^2.

The assignment in pairs and the electrical connection between feed terminal contacts allows the connection wires to be looped through to various terminal points. The current load from the power contact must not exceed 10 A for long periods. The current carrying capacity between two spring-loaded terminals is identical to that of the connecting wires.

Power contacts

On the right hand face of the Bus Terminal Controller there are three spring contacts for the power contact connections. The spring contacts are hidden in slots so that they cannot be accidentally touched. By attaching a Bus Terminal the blade contacts on the left hand side of the Bus Terminal are connected to the spring contacts. The tongue and groove guides on the top and bottom of the Bus Terminal Controllers and of the Bus Terminals guarantees that the power contacts mate securely.

3.2.3   Programming cable

Use the KS2000-Z2 programming cable for serial programming of the Bus Terminal Controller. This cable is included in the scope of supply of the KS2000 software, or it can be ordered separately (order identifier KS2000-Z2).

Fig. 11: Programming cable KS2000-Z2

KS2000-Z2

The programming cable offers the option of programming the BCxx50 via the serial interface.

NOTE
Do not interrupt the ground connection for the supply voltage
When the programming cable (between BCxx50 and PC) is connected, the ground connection of the Bus Terminal controller must not be interrupted or disconnected, since this may destroy the programming cable.

BECKHOFF

3.2.4 RS232 Connection

Nine-pin D sub

The cable is connected via a 9-pin D-Sub connector on the left side of the Bus Terminal Controller.
A three-core cable with shield should be used, with the shield being connected to the D-Sub connector casing.

Pin assignment of the RS232 D-Sub socket
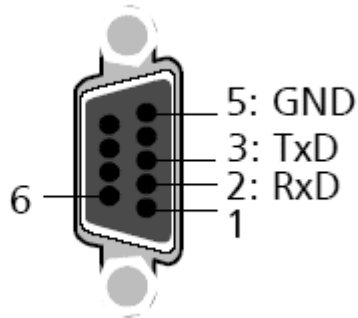
Fig. 12: RS232 pin assignment

Cable

The cable must be crossed type. GND is connected 1:1. Only use assigned and described pins.
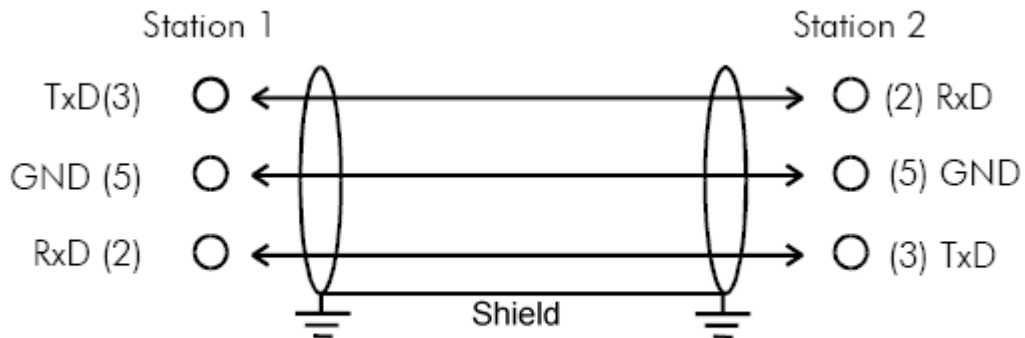
Fig. 13: Three-core cable with shield, crossed

3.2.5   RS485 connection

Nine-pin D sub

The cable is connected via a 9-pin D-Sub connector on the left side of the Bus Terminal Controller.
A two-core cable with shielding should be used. The shielding should be connected to signal ground (GND).

Pin assignment of the RS485 D-sub socket

8: RxD/TxD
6: 5V
5: GND
3: RxD/TxD
1

Fig. 14: RS485 pin assignment

Cable

The BC8050 can establish connections to several devices; the maximum permitted cable length is 1200 m.

Station 1   Station 2

TxD/RxD (3)   (3) RxD/TxD

RxD/TxD (8)   (8) TxD/RxD

Shield

Fig. 15: Two-core cable with shield

BECKHOFF

3.3 ATEX - Special conditions (extended temperature range)

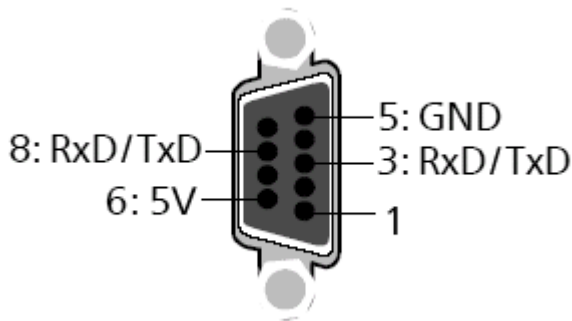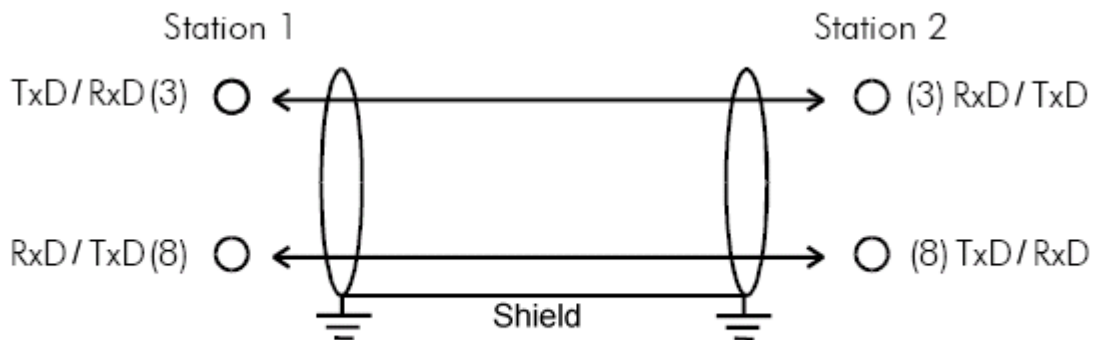| ⚠ **WARNING** |
|---|
| **Observe the special conditions for the intended use of Beckhoff fieldbus components with extended temperature range (ET) in potentially explosive areas (directive 2014/34/EU)!**

• The certified components are to be installed in a suitable housing that guarantees a protection class of at least IP54 in accordance with EN 60079-15! The environmental conditions during use are thereby to be taken into account!

• For dust (only the fieldbus components of certificate no. KEMA 10ATEX0075 X Issue 9): The equipment shall be installed in a suitable enclosure providing a degree of protection of IP54 according to EN 60079-31 for group IIIA or IIIB and IP6X for group IIIC, taking into account the environmental conditions under which the equipment is used!

• If the temperatures during rated operation are higher than 70°C at the feed-in points of cables, lines or pipes, or higher than 80°C at the wire branching points, then cables must be selected whose temperature data correspond to the actual measured temperature values!

• Observe the permissible ambient temperature range of -25 to 60°C for the use of Beckhoff fieldbus components with extended temperature range (ET) in potentially explosive areas!

• Measures must be taken to protect against the rated operating voltage being exceeded by more than 40% due to short-term interference voltages!

• The individual terminals may only be unplugged or removed from the Bus Terminal system if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!

• The connections of the certified components may only be connected or disconnected if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!

• The fuses of the KL92xx/EL92xx power feed terminals may only be exchanged if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!

• Address selectors and ID switches may only be adjusted if the supply voltage has been switched off or if a non-explosive atmosphere is ensured! |

Standards

The fundamental health and safety requirements are fulfilled by compliance with the following standards:

- EN 60079-0:2012+A11:2013
- EN 60079-15:2010
- EN 60079-31:2013 (only for certificate no. KEMA 10ATEX0075 X Issue 9)

Marking

The Beckhoff fieldbus components with extended temperature range (ET) certified according to the ATEX directive for potentially explosive areas bear the following marking:

II 3G KEMA 10ATEX0075 X Ex nA IIC T4 Gc Ta: -25 … +60°C

II 3D KEMA 10ATEX0075 X Ex tc IIIC T135°C Dc Ta: -25 ... +60°C
(only for fieldbus components of certificate no. KEMA 10ATEX0075 X Issue 9)

or

II 3G KEMA 10ATEX0075 X Ex nA nC IIC T4 Gc Ta: -25 … +60°C

II 3D KEMA 10ATEX0075 X Ex tc IIIC T135°C Dc Ta: -25 ... +60°C
(only for fieldbus components of certificate no. KEMA 10ATEX0075 X Issue 9)

3.4   Continuative documentation for ATEX and IECEx

NOTE
Continuative documentation about explosion protection according to ATEX and IECEx Pay also attention to the continuative documentation **Ex. Protection for Terminal Systems** Notes on the use of the Beckhoff terminal systems in hazardous areas according to ATEX and IECEx, that is available for download within the download area of your product on the Beckhoff homepage www.beckhoff.com!

4 Parameterization and Commissioning

4.1 Start-up behavior of the Bus Terminal Controller

When the Bus Terminal Controller is switched on it checks its state, configures the K-bus, creates a configuration list based on the connected Bus Terminals and starts its local PLC.
The I/O LEDs flash when the Bus Terminal Controller starts up. If the system is in an error-free state, the I/O LEDs should stop flashing after approx. 2-3 seconds. In the event of a fault the error type determines which LED flashes (see chapter *Diagnostic LEDs*).
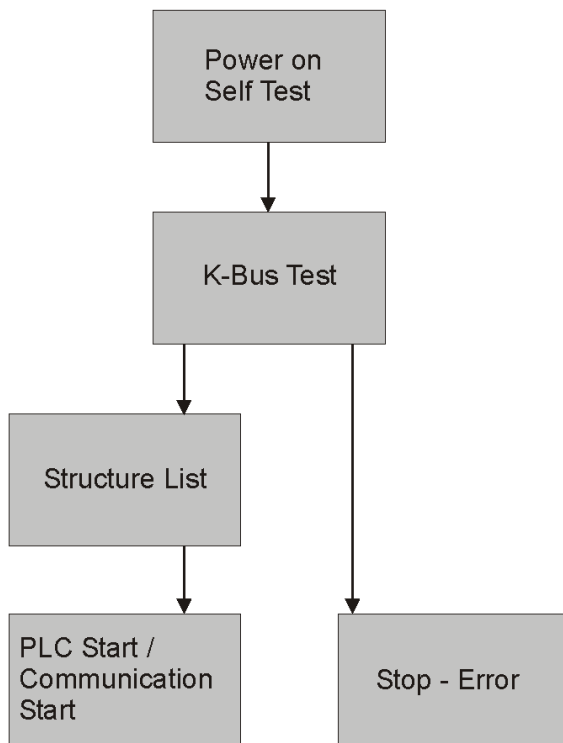

Fig. 16: Start-up behavior of the Bus Terminal Controller

4.2   Setting the Address

The address must be set via the two rotary selection switches. The default setting is 11.

The BC8150 Bus Terminal Controller supports different protocols.

Address 1 to 39

Automatic recognition of the following protocols

- Serial ADS, automatic baud rate detection, default 38400 baud, fixed data frame (8,e,1)
- KS8000 protocol, automatic baud rate detection, default 38400 baud, fixed data frame (8,e,1)
- KS2000 protocol, automatic baud rate detection, default 38400 baud, fixed data frame (8,e,1)
- ModbusASCII protocol, automatic baud rate detection, default 38400 baud, fixed data frame (8,e,1)
- TwinCAT PLC, protocol automatic baud rate detection default 38400 baud, fixed data frame (8,e,1)

Address 40 to 59

- ModbusRTU, configurable baud rate, default 9600, configurable data frame default 8,n,1

Changing the address

The switches can be set to the required position using a screwdriver. Ensure that the switches engage correctly. The lower switch is the ten-multiplier, the upper switch is the one-multiplier. The address change becomes active immediately.

Example

You want to set address 34:

- Lower rotary selection switch Sx11: 3
- Upper rotary selection switch Sx10: 4
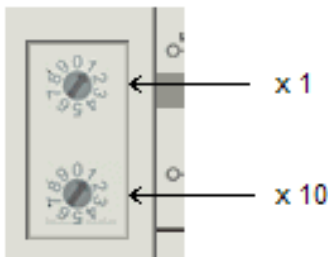
Fig. 17: Setting the node ID

Special functions using the address switch

In addition, the address switch can be used for following functions. To this end the associated address has to be set (as long as the coupler is switched off), only the end terminal may be connected (KL9010), and the coupler must then be connected to the supply voltage.

Address 99: Factory setting
Address 98: Delete boot project
Address 97: Delete TwinCAT configuration

4.3   Baud rate

The Bus Coupler features automatic baud rate detection. When the supply voltage is switched on, the Bus Coupler searches for the current baud rate and stores it as long as the supply voltage is present.

> **ⓘ**   **Restarts the bus coupler for automatic baud rate detection**
> If the master baud rate is changed, the Bus Coupler has to be switched off and on again in order to restart automatic baud rate detection.

Linear extension of RS 232

Baud rate in kbit/sec	9,6	19,2	38,4
Permissible cable length in m	15	15	15

4.4   Configuration

4.4.1   Overview

Configuration types

The Bus Terminal controllers of the BCxx50, BCxx20 and BXxx00 series can be configured in two different ways: DEFAULT CONFIG or TwinCAT CONFIG.

DEFAULT-CONFIG

Bus Terminals are mapped in the order they are inserted, i.e. first the complex Bus Terminals followed by the digital Bus Terminals.

The complex Bus Terminals are mapped as follows:

- Word Alignment
- complex representation

⚠ CAUTION
The process image depends on the connected terminals!
The process image changes when a terminal is added or removed!

The data of the fieldbus slaves interface are referred to as PLC variables. The PLC variables have addresses from %QB1000 and %IB1000

The DEFAULT CONFIG (without PLC program) can also be used for writing and testing of the Connected Bus Terminals. To this end, the Bus Terminal Controller must be scanned in the System Manager, and FreeRun mode must be enabled (to use this function, no PLC program may be active on the Bus Terminal Controller).

TWINCAT-CONFIG

In the TwinCAT CONFIG the Bus Terminals and PLC variables can be freely linked as required (TwinCAT System Manager file required). The configuration is transferred to the coupler via the System Manager and ADS.

The following is required for the TwinCAT configuration (TC file):

- Via the fieldbus (PROFIBUS, CANopen, Ethernet)
  PROFIBUS: (BC3150, BX3100)
  - PC with FC310x from version 2.0 and TwinCAT 2.9 build 1000
  - BX3100 with CIF60 or CP5412

- ◦ TwinCAT 2.9 build 946
  (**NOTE:** with PROFIBUS cards from Hilscher only one ADS communication is permitted, i.e. either System Manager or PLC Control)
  CANopen: (BC5150, BX5100)
- ◦ PC with FC510x from version 1.76 TwinCAT build 1030
  DeviceNet: (BC5250, BX5200)
- ◦ on request
  Ethernet: (BC9050, BC9020, BC9120, BX9000)
- ◦ PC with TwinCAT 2.10 build 1322
- Via the serial ADS TwinCAT 2.9 build 1010
  - ◦ BX3100 version 1.00
  - ◦ BX5100 version 1.00
  - ◦ BX5200 version 1.10
  - ◦ BX8000 version 1.00
  - ◦ BC3150, BC5150, BC5250, BC9050, BC9020, BC9120 from firmware B0
  - ◦ For BC8150 from TwinCAT 2.10 build 1243

BCxx50 and BXxx00 can be parameterized via the System Manager of the TwinCAT program.

- Variable I/O mapping
- Type-specific PROFIBUS data (BC3150 and BX3100 only)
- RTC (real-time clock) (BX series only)
- SSB (Smart System Bus) (BX series only)
- PLC settings
- K-Bus settings

The configuration can be transferred to the BCxx50 or BXxx00 via fieldbus ADS protocol or serial ADS protocol.

The TwinCAT configuration can be used to link variables, I/Os and data. The following is possible:

- PLC - K-BUS
- PLC fieldbus (e.g. PROFIBUS slave interface to PLC)
- K-bus fieldbus (only for BX controllers)
- Support for TwinSAFE terminals (only BX controllers from firmware 1.17)

In addition, the TwinCAT configuration can be used to parameterize special behavior, for example whether data are preserved or set to "0" in the event of a fieldbus error.
The real-time clock can be set via a tab in the system manager.

Work steps

1. Setting the fieldbus address
2. Open the System Manager and create a TC file
3. Configure fieldbus data in the TC file
4. Save the TC file
5. Opening a new system manager, creating a PC file and reading in saved TX file
6. Creating a link to a PLC task
7. Saving the configuration
8. Starting the TwinCAT system
9. Open the TC file in the System Manager, complete the configuration and transfer it to the BCxx50, BCxx20 or BXxx00
10. Transfer the program to BCxx50, BCxx20 or BXxx00
11. Creating a boot project

4.4.2   Creating a TwinCAT configuration

In order to configure a Bus Terminal Controller of the BCxx50, BCxx20, BXxx00 or BC9191 series, create a BX file in the System Manager. To simplify matters, files for the basic units have already been prepared. Open the corresponding Bus Terminal Controller with *New from Template*.

Fig. 18: Creating a TwinCAT configuration

Select the corresponding Bus Terminal Controller.
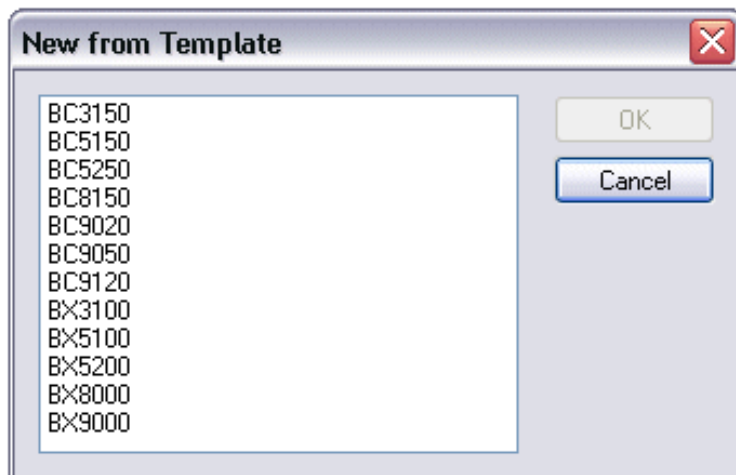
Fig. 19: Selecting the Bus Terminal Controller

All Bus Terminal Controller components are now available:

- Fieldbus interface
- K-bus interface [▶ 40]
- PLC program
- SSB (only Bus Terminal Controllers of the BX series)

Please refer to the relevant chapter for device configuration.

4.4.3   Downloading a TwinCAT configuration

The TwinCAT configuration is loaded into the Bus Terminal Controller via ADS protocol.

Serial ADS protocol

(all Bus Terminal Controllers of the BXxx00 and BCxx50 series)

Enter the serial ADS connection, as described in the chapter Serial ADS [▶ 36].

ADS protocol via the fieldbus

(BC3150, BC5150, BC9x20, BC9050, BX3100, BX5100, BX9000, BC9191 only)

A prerequisite is that TwinCAT operates as master and is engaged in data exchange, i.e. the physical and fieldbus configuration must be complete, and data exchange must take place between the master (e.g. fieldbus master card) and the Bus Terminal Controller.

Choose Target System

Select the Bus Terminal Controller onto which the configuration is to be loaded. Use the function key F8 to open the dialog for downloading your file to the corresponding device.
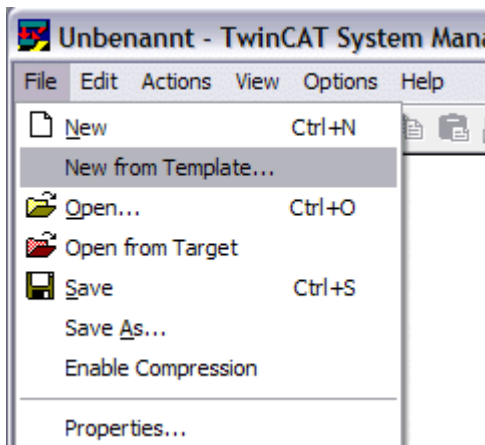

Fig. 20: Downloading a TwinCAT configuration

Select the corresponding Bus Terminal Controller.

Fig. 21: Selecting the Bus Terminal Controller
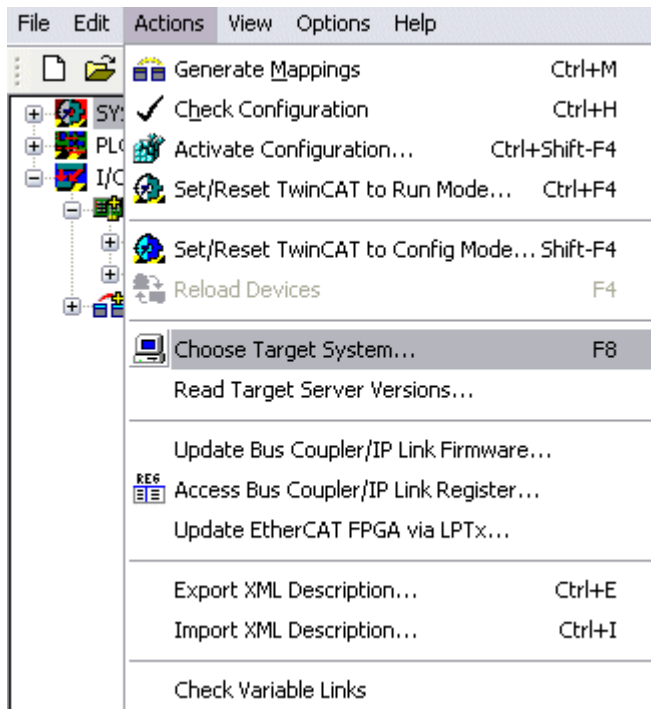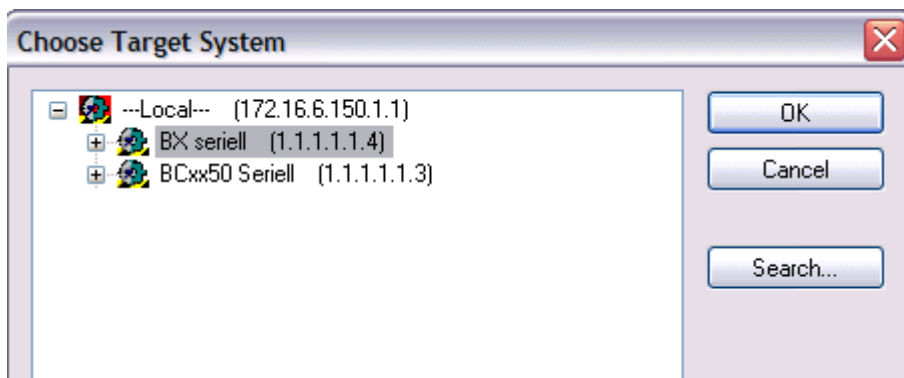
The state of the Bus Terminal Controller is shown at the bottom right of the System Manager.

Fig. 22: State of the Bus Terminal Controller

In *Config mode / FreeRun* the configuration can now be downloaded to Bus Terminal Controller. If the Bus Terminal Controller is in *Stop mode*, ADS communication is not yet activated. In this case, it is not possible to download the configuration.

To activate the TwinCAT configuration select Ctrl+Shift+F4 or *Activate Configuration*.
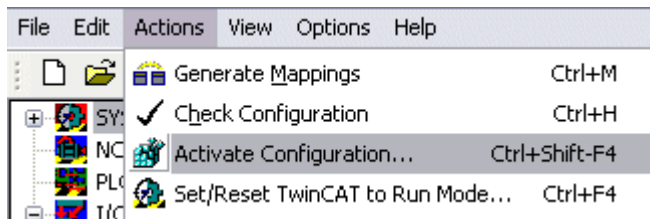
Fig. 23: Activating the TwinCAT configuration

The current configuration is loaded onto the Bus Terminal Controller. The display will show *Store Config*, and the BUS and I/O LED will flash. Once the configuration is successfully loaded onto Bus Terminal Controller, *TwinCAT Config* should appear in the display of a BXxx00. The corresponding program can now be transferred to the Bus Terminal Controller (program-download via the fieldbus).

4.4.4   Uploading a TwinCAT configuration

The TwinCAT configuration is loaded into the Bus Terminal Controller via ADS protocol.

Serial ADS protocol

(all Bus Terminal Controllers of the BCxx50, BCxx20 and BXxx00 series)

Enter the serial ADS connection, as described in the chapter Serial ADS [▶ 36].

ADS protocol via the fieldbus

(BC3150, BC5150, BC9x20, BC9050, BX3100, BX5100, BX9000, BC9191 only)

A prerequisite is that TwinCAT operates as master and is engaged in data exchange, i.e. the physical and fieldbus configuration must be complete, and data exchange must take place between the master (e.g. fieldbus card) and the Bus Terminal Controller.

Choose Target System

Select the Bus Terminal Controller onto which the configuration is to be loaded. Use the function key [F8] to open the dialog for downloading your file to the corresponding device.
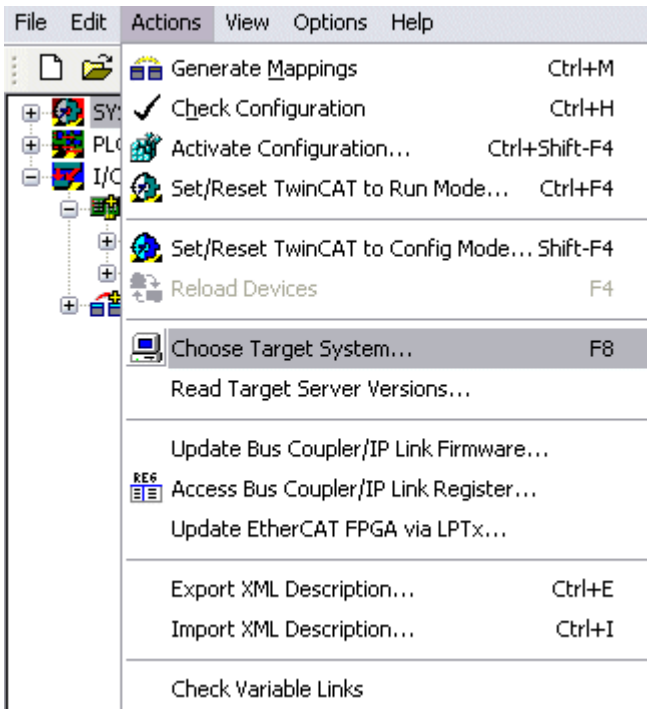
Fig. 24: Choose Target System

Select the corresponding Bus Terminal Controller.
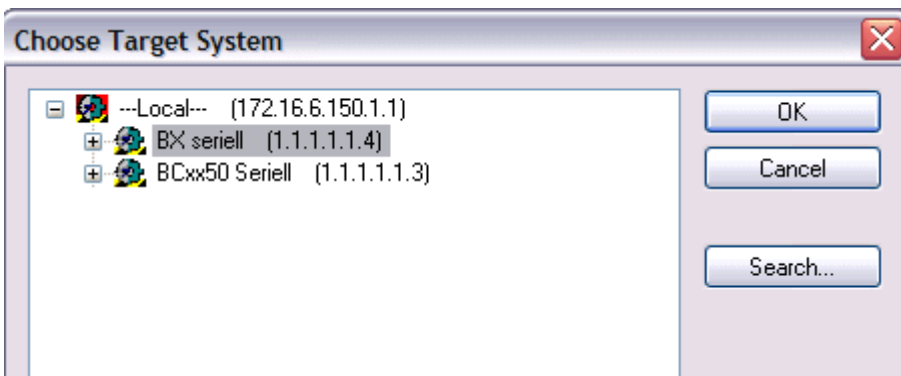
Fig. 25: Selecting the Bus Terminal Controller

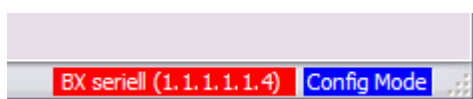The state of the Bus Terminal Controller is shown at the bottom right of the System Manager.
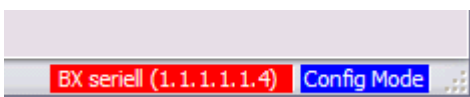
Fig. 26: State of the Bus Terminal Controller

Click on the red folder. The TwinCAT configuration will now be uploaded.
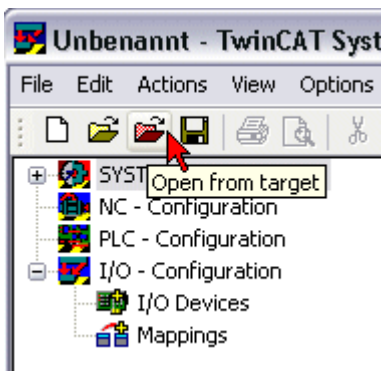
Fig. 27: Uploading the TwinCAT configuration

4.4.5 Resources in the Bus Terminal Controller

The memory resources assigned in the Bus Terminal Controller are shown in the System Manager in the *Resources* tab of the Bus Terminal Controller.

Mapping code

The mapping code is required for calculating the TwinCAT configuration (see Figure *Memory for the code mapping*). The percentages are added here. In the example from Fig. *Memory for code mapping*, 8% of the memory is allocated to the mapping calculation.
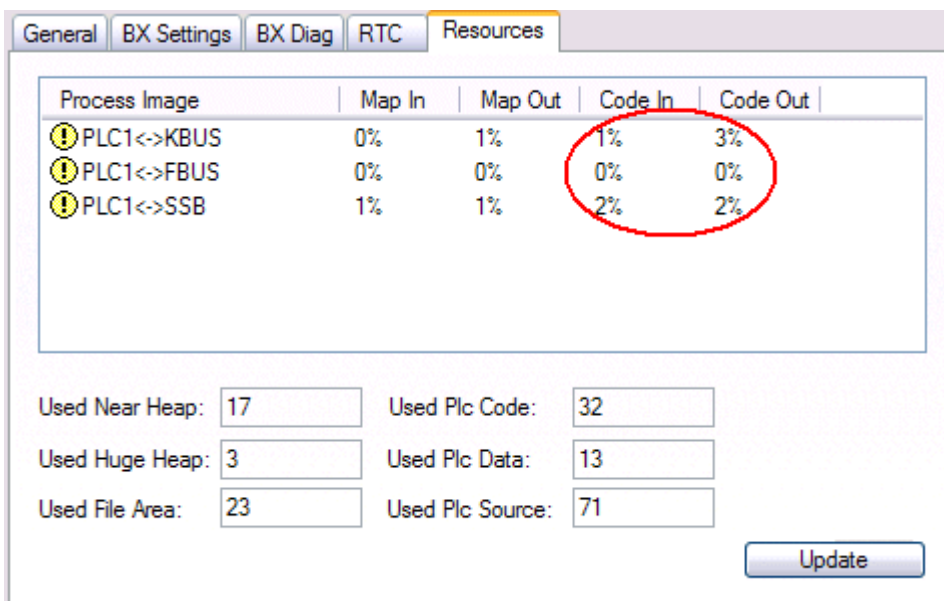
Fig. 28: Memory for code mapping

Data memory mapping

Data memory for mapping. The values are to be considered individually, i.e. each value can be up to 100%.
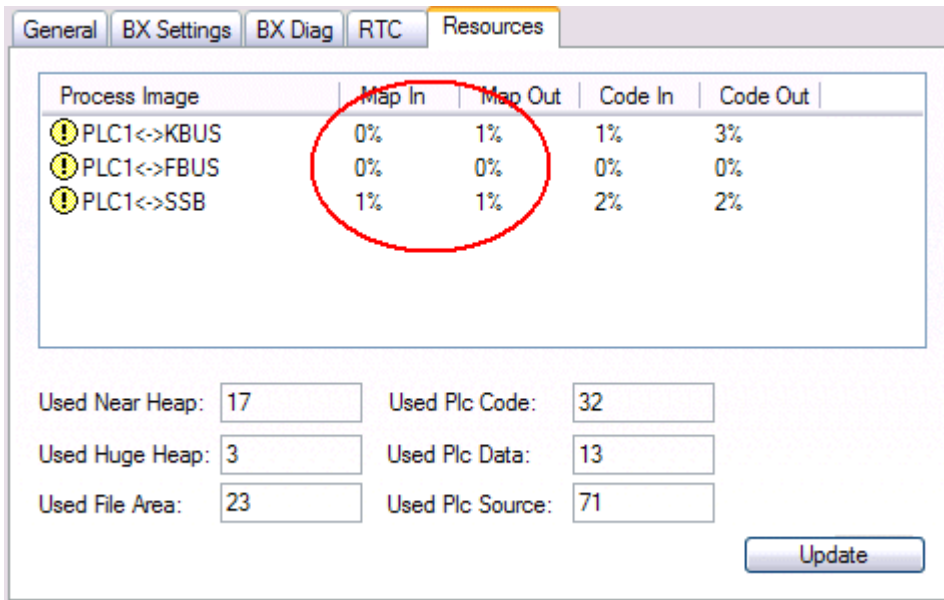
Fig. 29: Data memory mapping

Used code and data memory

Fig. *Code and data memory* (1) "Used PLC code" in %.
Fig. *Code and data memory* (2) "Used PLC data" in %.
Fig. *Code and data memory* (3) "Used PLC source" in %.
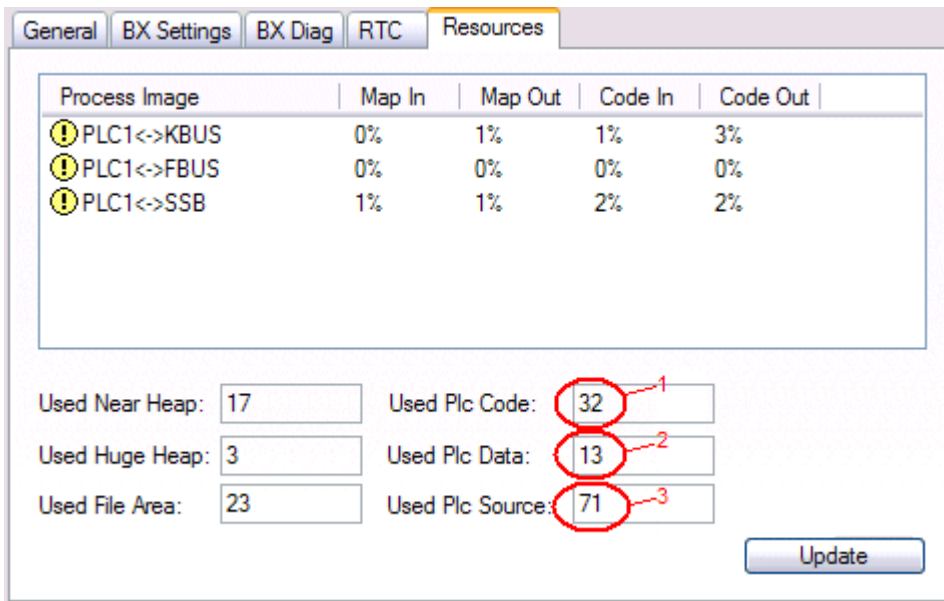
Fig. 30: Code and data memory

Other memory

Fig. *Other Memory* (1) "Used Near Heap" is required for the COM interface and SSB. % values.
Fig. *Other Memory* (2) "Used Huge Heap" is required for the ADS communication. % values. This value should be less than 30 %.
Fig. *Other Memory* (3) "Used File Area" is required for the TwinCAT configuration, the TSM file and the 16 kbyte flash access. % values.

Fig. 31: Other memory

4.4.6 ADS connection via serial interface

(from firmware version 1.xx or 0.99x, Bus Terminal Controllers of the BX series and for all BCxx50)

From TwinCAT 2.9 build 1020 (TwnCAT level PLC, NC or NCI)

i ● **Use only a serial connection**

To ensure trouble-free operation of the ADS link via the serial interface, only a serial connection to the BX controller is allowed.
After successful configuration via the System Manager, close the System Manager before starting programming.

i ● **AMS Net ID in delivery state (default)**

For BX9000

The default AMS Net ID is 172.16.21.20.1.1. If the IP address of the BX9000 is changed, the AMS Net ID of the BX9000 also changes. There is a menu option for displaying the current AMS Net ID. Example: If you change the IP address to 10.2.3.7, the AMS Net ID changes to 10.2.3.7.1.1.

For BC9050, BC9020, BC9120

The default AMS Net ID is 172.16.xxx.[DIP switch].1.1. If the IP address of the BX9xxx is changed, the AMS Net ID of the BX9xxx also changes.
Example: If you change the IP address to 10.2.3.7, the AMS Net ID changes to 10.2.3.7.1.1.
BC9050: DEFAULT 172.16.21.[DIP-Switch].1.1
BC9020: DEFAULT 172.16.22.[DIP-Switch].1.1
BC9120: DEFAULT 172.16.23.[DIP-Switch].1.1

Initializing the ADS connection

Enter the Bus Terminal Controller in the remote connection under TwinCAT. Click on the TwinCAT icon and open the features menu. The following settings can be made under the >AMS Remote< tab.
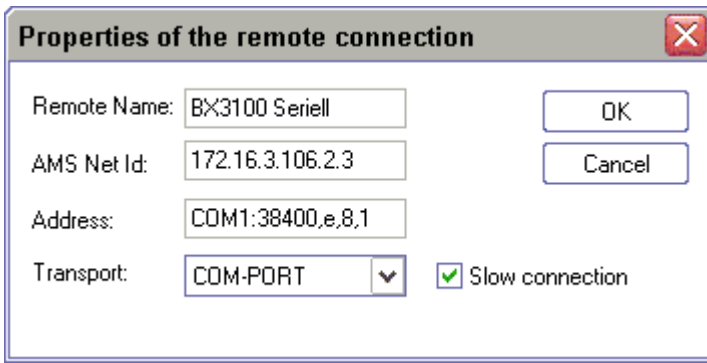
Fig. 32: Properties of the remote connection

Remote Name: Any
AMS-Net-ID: 1.1.1.1.1.1 (Default)
Address: COM Port: Baud rate, parity, data bits, stop bits
Transport: Select "COM port"

When the Bus Terminal Controller is switched on, the default AMS Net ID is always "1.1.1.1.1.1" (except all Ethernet Controllers).
The AMS Net ID can be changed as required. Please note that the new AMS Net ID cannot be changed again in this way.

If you need to change the new AMS Net ID again, you have to restart the Bus Terminal Controller, so that the AMS Net ID is reset to the default AMS Net ID, "1.1.1.1.1.1".
You can now change the AMS Net ID again.

●
ℹ

Strings can only be entered at the second call

No strings can be entered under address when the dialog is first called (see above). Enter the name, AMS Net ID and transport type and close the dialog. With the second call you can enter your COM port.

The communication starts when TwinCAT is in Config mode (TwinCAT icon is blue) or RUN mode (TwinCAT icon is green). The COM interface remains open until a TwinCAT stop occurs (TwinCAT icon is red). It is then available again for other programs. No error message is issued if the COM interface is used by another program during a TwinCAT restart (e.g. by the KS2000 configuration software).

●
ℹ

AMS Net ID after ADS connection via the fieldbus

If you have addressed the Bus Terminal Controller with an ADS connection via the fieldbus before the serial ADS was used, the AMS Net ID was automatically changed by the System Manager. In this case a new serial ADS connection is only possible, if the AMS Net ID is adjusted.

BX series: reading the AMS Net ID

The current AMS Net ID can be read from the menu via the display of BX series Bus Terminal Controller.

| **AMS** | AMS Net ID |
| 1.1.1.1.1.1 | |

4.5 RS232 interface

4.5.1 TwinCAT Configuration - RS232 Interface

TwinCAT 2.10 Build 1243 or higher is required for using the TC configuration.

Open the BC8150 template project.
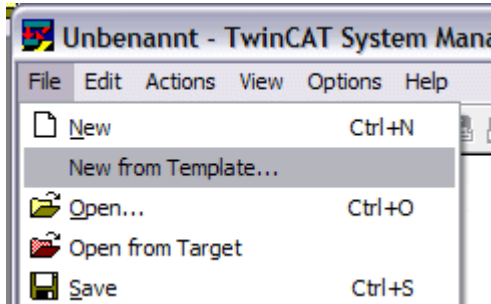
Fig. 33: Selection dialog "New from Template"

Select BC8150.
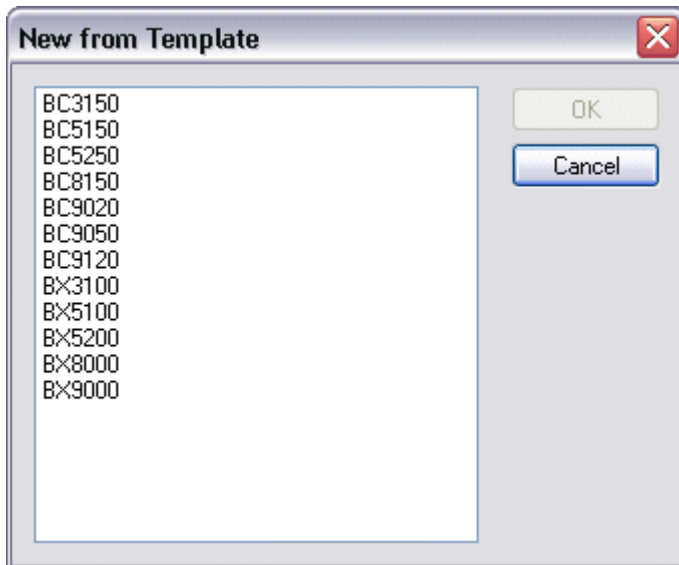
Fig. 34: Selection of Bus Controller

Open the device (BC8150) and click on BC8150. On the right in the System Manager there are two tabs, i.e. *General* and *Communication Properties*. Select *Communication Properties*. The following settings are available. Watchdog setting - at 0 ms the watchdog is disabled, adjustable up to 64000 ms max. The baud rate and parity are only valid if the address switch is set to 40-59. The input and output sizes are calculated from the attached variables.
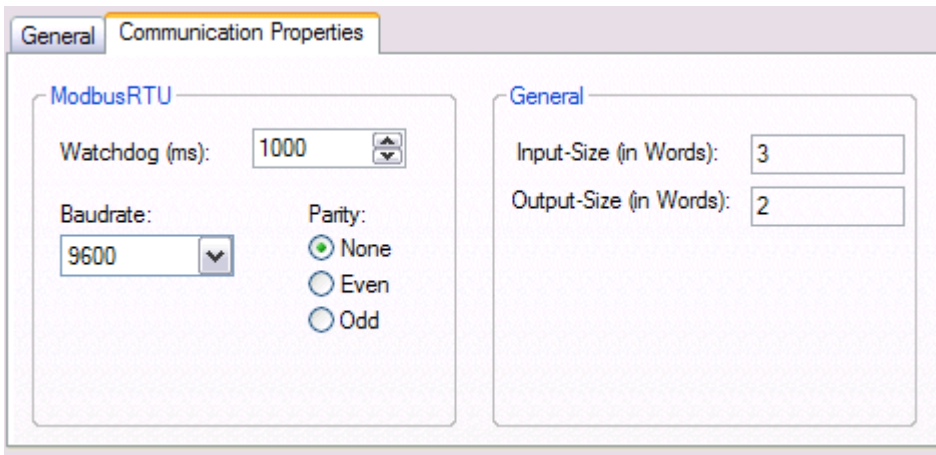
Fig. 35: "Communication Properties" tab

Open the BC8150 and right-click on *Inputs* for setting up inputs or *Outputs* for setting up outputs to be received or sent via the RS232 interface.
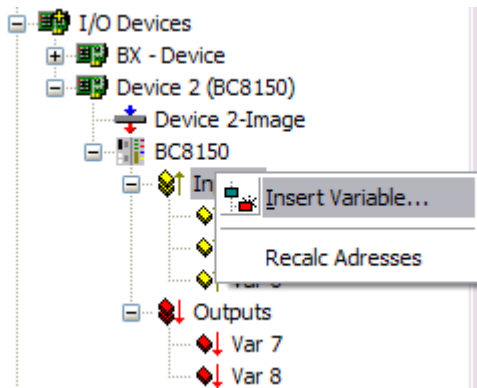
Fig. 36: Creating variables (inputs, outputs)

Please not that an even start address (0, 2, 4, 6, ...) has to be selected for variables with a size of 2 bytes or greater.
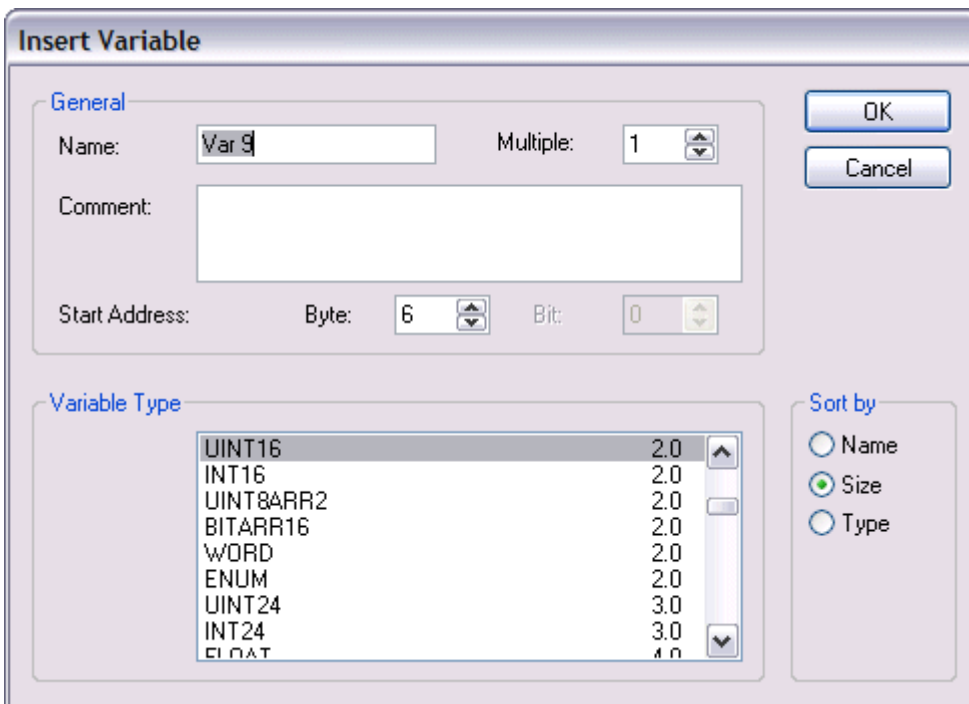
Fig. 37: Definition of variables

4.6   K-bus

> **ℹ**   **Bus Terminal and end terminal required**
>
> To operate a Bus Terminal Controller of the BC or BX series, at least one Bus Terminal with process image and the end terminal must be connected to the K-bus.

BX Settings tab
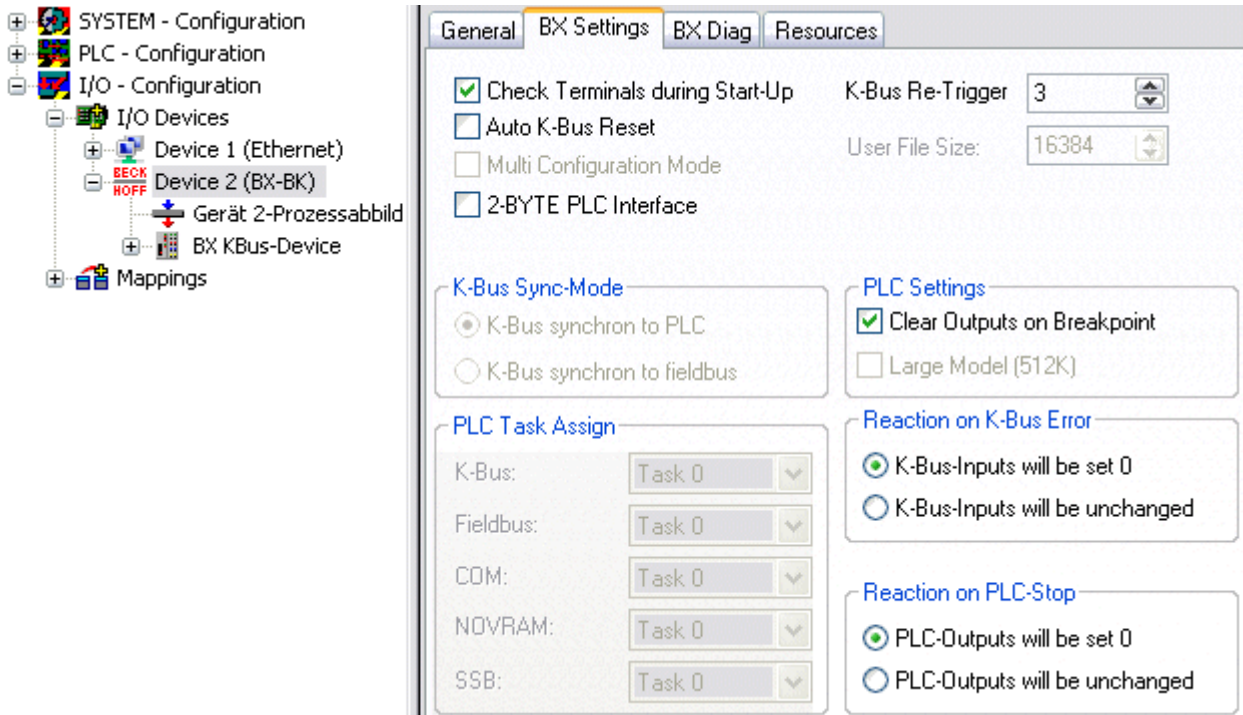
Fig. 38: BX Settings tab

Check Terminals during Start-up

When a boot project is created, the current Bus Terminal configuration is stored. The connected Bus Terminals are checked when the Bus Terminal Controller restarts. If this option is selected, the Bus Terminal Controller does not enter into data exchange. The PLC project will not be started.

Auto K-Bus Reset

Once a K-bus error has been rectified, the Bus Terminal Controller automatically resumes the data exchange.

⚠ CAUTION
Once a K-Bus error has been rectified, the outputs become active again immediately!
Ensure that the outputs are reactivated immediately and that analog outputs retain their programmed value, if this is not dealt with in your PLC project.

Clear Outputs on Breakpoint

If breakpoints are set in PLC Control, the K-Bus is no longer processed, i.e. the outputs are set to a safe state (zero).

K-Bus Sync Mode

Writing and reading of the Bus Terminals can take place synchronously with task 1 or the fieldbus.

K-Bus Re-Trigger

If the processor is busy dealing with the PLC project or the SSB, the K-Bus cannot be processed for a certain amount of time. This leads to triggering of the Bus Terminal watchdog and dropping of the outputs. The Bus Terminal Controller is set such that the K-bus watchdog is re-triggered 3 times after 85 ms. The K-Bus watchdog would then be activated.
K-Bus Re-Trigger 0: 100 ms
K-Bus Re-Trigger 1: 2 x 85 ms = 170 ms
K-Bus Re-Trigger 2: 3 x 85 ms = 255 ms
K-Bus Re-Trigger 3: 4 x 85 ms = 340 ms

Reaction on K-Bus Error

In the event of a K-Bus error, the K-Bus inputs are set to "0" or retain their last state.

Response on PLC-Stop

The user can set the behavior of the fieldbus output data in the event of the PLC project being stopped. The master will use these data as input data. In the event of a PLC stop, the data can be set to "0" or remain unchanged.

BX Diag tab

Display of the cycle time for task 1, K-bus, fieldbus processing and the SSB load.
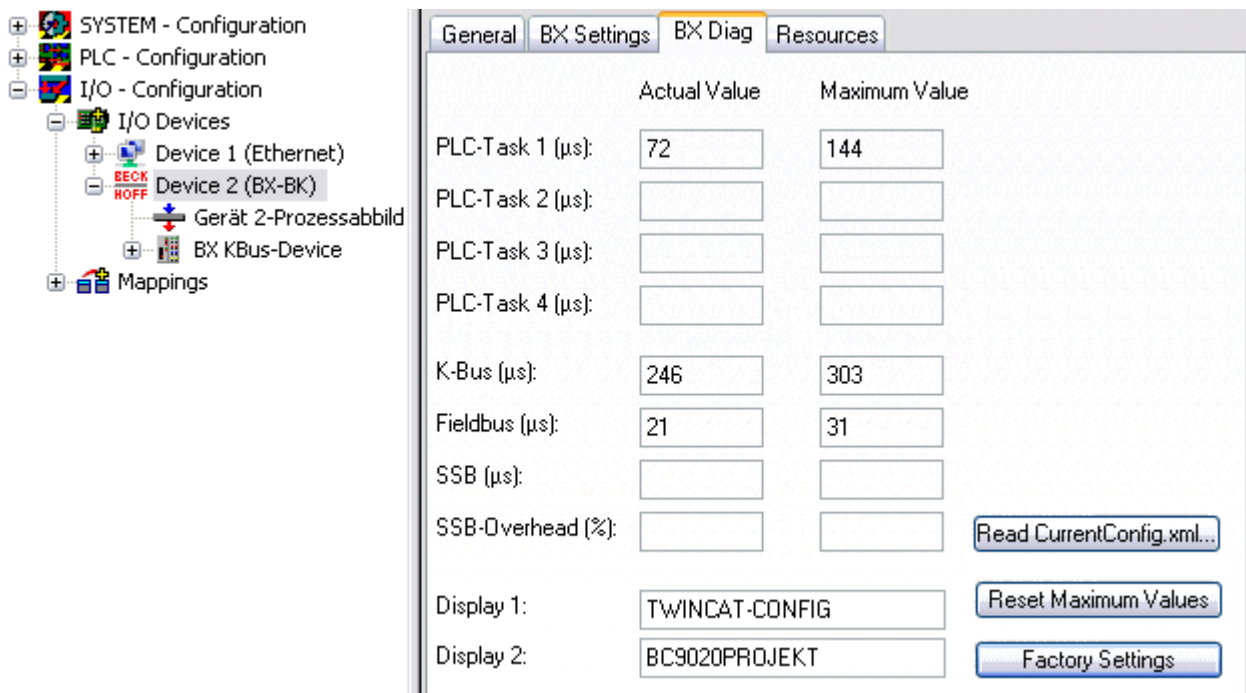
Fig. 39: BX Diag tab

Factory Settings: the Bus Terminal Controller is set to its delivery. These settings are reactivated via Restart System or by switching the system off and on again (display shows DEFAULT-CONFIG).
Reset Maximum Values: resets the maximum values

K-Bus variables

```
⊟ ⬚ Box 2 (CX1100-KB)
   ⊟ ◈↑ Inputs
        ◈↑ PlcInterface
        ◈↑ KBus-State
   ⊞ ◈↓ Outputs
   ⊞ ⬚ Term 2 (KL1032)
   ⊞ ⬚ Term 3 (KL1032)
```
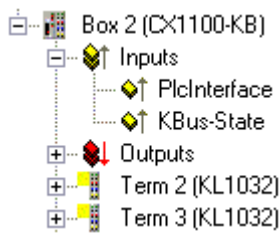
PLC interface: Not supported (only included for moving CX or BX projects)

K-bus state: see Diagnostics

4.7 Configuration software KS2000

Bus Terminal controllers of the BCxx50, BXxx20 and BXxx00 series cannot be parameterized and configured with the KS2000 configuration software. These devices must be configured with the TwinCAT System Manager.

The KS2000 configuration software offers configuration and diagnostic support for the Bus Terminals attached to the Bus Terminal Controller.
It is advisable to set the baud rate in the KS2000 configuration software and the BCxx50 BCxx20 and BXxx00 to 38400 baud (8 data bits, even, 1 stop bit).

> **ⓘ** **COM1 - automatic baud rate detection**
>
> The COM 1 interface of the BXxx00 features automatic baud rate detection between 9.6 kbaud and 56.4 kbaud.

> **ⓘ** **Required KS2000 version**
>
> Configuration or diagnostics of Bus Terminals at BXxx00 is supported from KS2000 version 4.3.14.

In some Bus Terminals (e.g. KL25xx, KL6811, KL6201, KL6401) the following parameters must be set in order to be able to use the configuration dialogs:

- A PLC project or boot project must be deactivated.

- The BX controller must be in its default configuration. Set the manufacturer's setting or switch to Config Mode in the TwinCAT System Manager (blue TwinCAT icon).

- The BX controller must be in FreeRun mode. Activate it with the TwinCAT System Manager.

You can now log in with the KS2000 configuration software via ADS (port 100) or the serial cable and use the KS2000 dialogs in the Bus Terminals.

5   Programming

5.1   BCxx50 PLC features

Description	Value
Data memory	32 kbyte
Program memory	48 kbyte minus task-configuration minus POUs during online change
Source code memory	128 kbyte
RETAIN	2 kbyte
INPUT	2 kbyte
OUTPUT	2 kbyte
FLAG	4 kbyte
Max. variable size	16 kbyte
Max. POUs	Limited by memory

5.2   TwinCAT PLC

The Beckhoff TwinCAT Software System turns any compatible PC into a real-time controller with a multi-PLC system, NC axis control, programming environment and operating station. The TwinCAT programming environment is also used for programming the BC/BX. If you have TwinCAT PLC (Windows NT4/2000/XP) installed, you can use the fieldbus connection or the serial port for downloading and debugging software.

TwinCAT I/O or TwinCAT PLC can also be used as the Ethernet Master (host), in order to exchange process data with the Bus Terminal Controller. TwinCAT provides you with the System Manager as a configuration tool, as well as the drivers and the ADS protocol.

Bus Terminal Controllers of the BCxx50, BCxx20 and BXxx00 series

These 2nd-generation Bus Terminal Controllers are configured with the TwinCAT System Manager and programmed with TwinCAT PLC Control. TwinCAT PLC must be installed for these couplers (Windows NT4, Windows 2000, Windows XP).

Programming and program transfer

- via the serial interface [▶ 63]
- via the fieldbus interface (only for Bus Terminal controllers for PROFIBUS, CANopen and Ethernet)

Online change

The Bus Terminal Controllers of the BX series and the BCxx50 support online change. This means that the PLC program is replaced with a new program without interrupting the program. The switch-over to the new program occurs after the task is completed. This means that two versions of the PLC program have to be stored. 512 kbyte are available, which therefore have to be divided by two, leaving 256 kbyte for the actual PLC program. In addition, several kbyte are required for task configuration etc. During an online change, dynamic data are stored in memory. Should a program approach the memory limit (program size greater than 240 kbyte), the online change may no longer work, even though the program may still be written to the BX after "Rebuild all".

When is online change not available?

Online change is not available under certain conditions,.

- Inserting of a new library
- Changing the task setting

BECKHOFF

- "Rebuild all"
- Controller memory limit is almost reached (PLC program greater than 90%)

5.3   TwinCAT PLC - Error codes

Error type	Description
PLC compiler error	Maximum number of POUs (...) exceeded
PLC compiler error	Out of global data memory ...

Error POUs

For each function block one POU (process object unit) is created. 256 function blocks are available by default.

Error 3612: Maximum number of POUs (100) exceeded! Compile is aborted.
Data allocation
1 Error(s), 0 Warning(s).

Fig. 40: Maximum number of POUs exceeded

If libraries are integrated this value may be insufficient. In this case, the number of POUs should be increased.

To this end, open in PLC Control under Projects/Options...
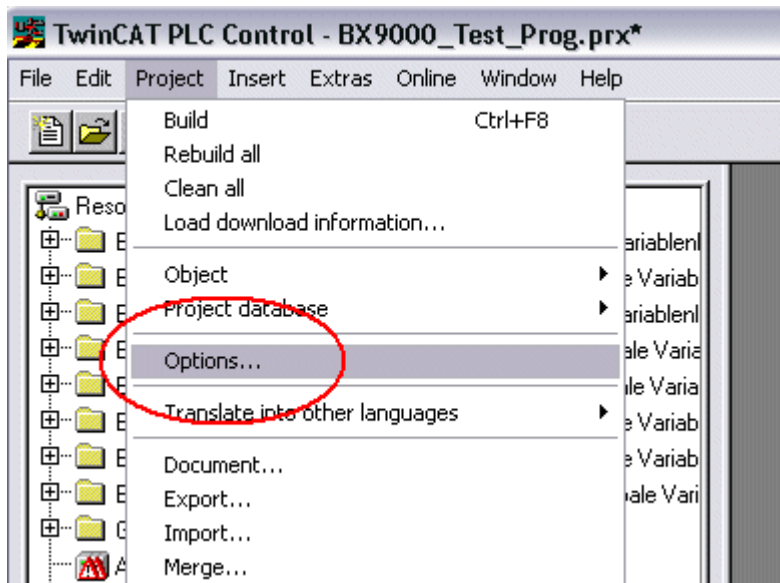
Fig. 41: Menu path Projects / Options / Controller Settings

...the controller settings.
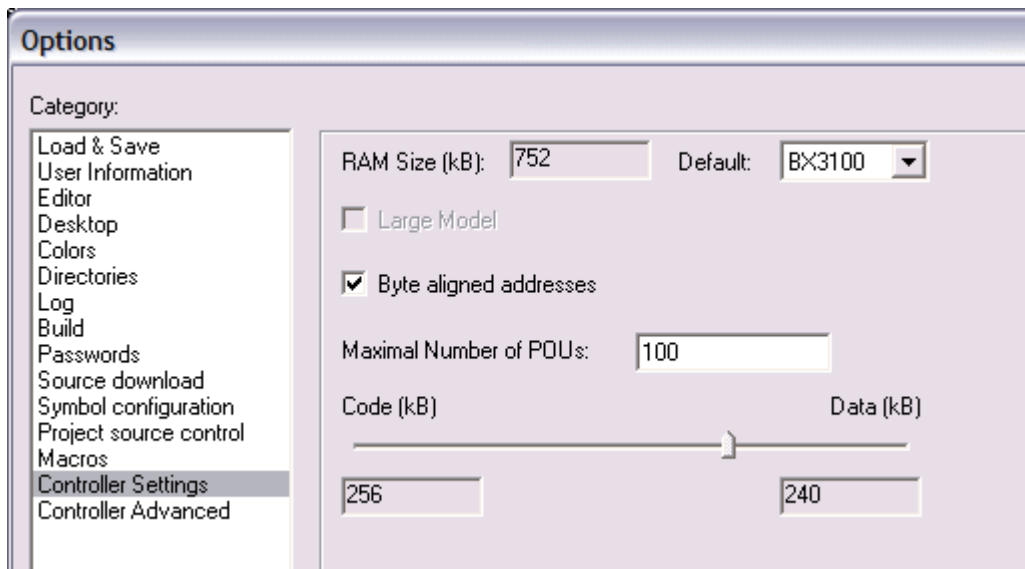
Fig. 42: Controller settings

Changing these settings will deactivate online changes.

Global memory error

Fig. 43: Global memory insufficient

2 x 16 kbyte of data are available by default. If large data quantities are to be used, this range should be increased. A maximum of 14 data segments are possible for the BX.
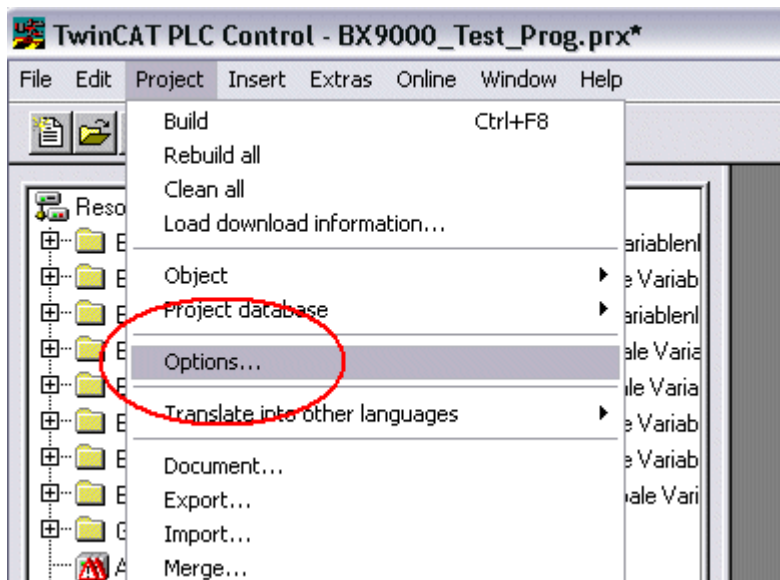
Fig. 44: Menu path Projects / Options / Build

BECKHOFF
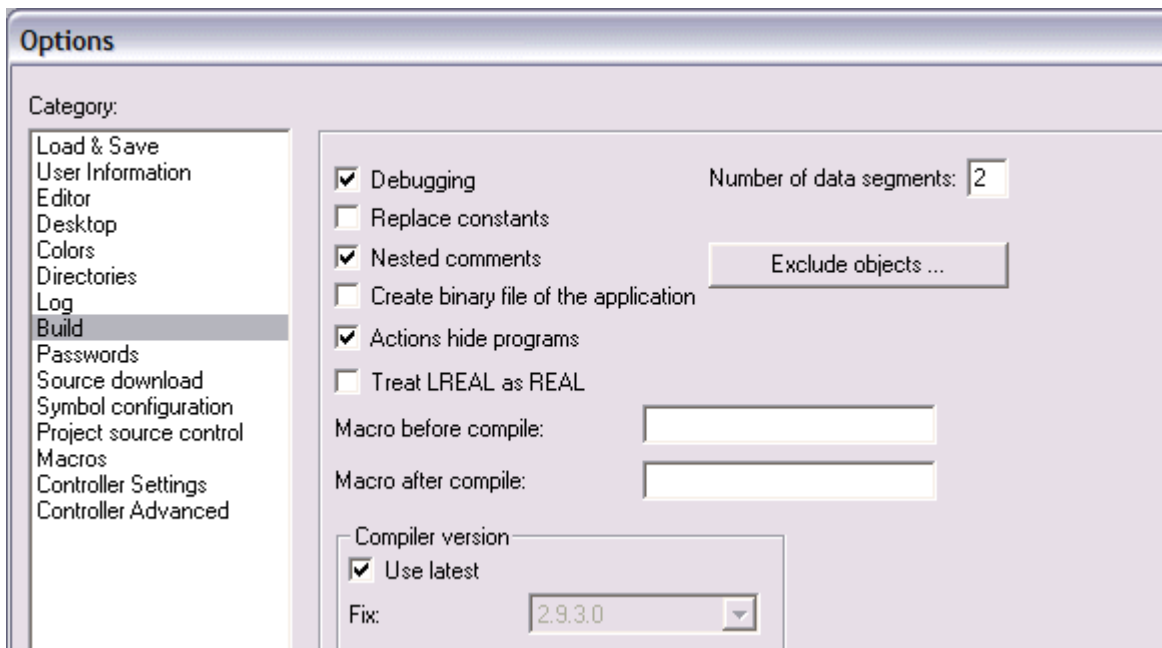
Fig. 45: Build

5.4   Remanent data

2000 bytes of remanent data are available on the BC9191 and the BX controller. These data are declared as VAR RETAIN in PLC Control.

Example

```
VAR RETAIN
    Test    :BOOL;
    Count   :INT;
END_VAR
```

Retain data are located between VAR RETAIN and END_VAR. These data are stored in a NOVRAM and are consistent across the whole 2 kbyte range. The RETAIN data are stored in the NOVRAM after each cycle. For 2 kbyte approx. 2 ms are required (for 1 kbyte approx. 1 ms). The variables can be configured locally or globally. Allocated variables (%MB, %QB, %IB) cannot be used as remanent data.

> ● **Do not use VAR_RETAIN in function blocks**
>
> **i** VAR_RETAIN should not be used in function blocks. All FB data are copied into the retain memory. This leads to an unnecessary increase in cycle time, and the retain memory is filled with unnecessary data.

> ● **Do not use variables with address as remanent data**
>
> **i** Variables that have been assigned an address (%MB, %QB, %IB) must not be used as remanent data.

Example for remanent data in the function block

This should be avoided, if possible, since all the data of a function block, in which even just a single remanent bit is found, are stored by default. A program sample can be found below.

Function block test (no program code required - in ST semicolon is sufficient)

```
FUNCTION_BLOCK Test
VAR_INPUT
END_VAR
VAR_OUTPUT
END_VAR
VAR
END_VAR
VAR_IN_OUT
    Counter   :INT;
END_VAR
```

MAIN program

```
PROGRAM MAIN
VAR
    fb_Test:Test;
END_VAR
VAR RETAIN
    iCounter1:INT;
END_VAR

fb_Test(Counter:=iCounter1);
```

5.5 Allocated flags

4 kbyte of allocated flags are available. They can be used to assign different variable types to the same address, e.g. for converting strings to bytes. Data can also be placed here that can be read or written via ADS by the controller.

> ● **Allocated variables are not remanent data**
>
> For the Bus Terminal Controllers of the BX series and the BCxx50 the allocated variables are **not** saved as remanent data.

Reading/writing of allocated flags via ADS

The flags may also be read via the controller and ADS. In PROFIBUS, the DPV-1 services are used for this purpose, in CANopen SDO communication is used.
The AmsNetID can be obtained from the System Manager, or it can be displayed via the Bus Terminal Controller menu.
The PLC port number is 800.

Index group	Meaning	Index offset (value range)
0x4020	Flag (only BXxxx0)	0..4096

Example

BX program

```
VAR
    Flag_01 AT %MB0: WORD;
END_VAR
```

TwinCAT PC/CX master program

```
VAR
    fbADRSREAD: ADSREAD;
    Flag_M: WORD;
END_VAR
```

```
fbADRSREAD(
    NETID:='172.16.3.0.2.3' ,   (* AMSNetId BX *)
    PORT:=800 ,                  (* 800 - PLC *)
    IDXGRP:=16#4020 ,            (* 0x4020hex falgs *)
    IDXOFFS:=0 ,                 (* byte offset *)
    LEN:=2 ,                     (* Lenght byte *)
    DESTADDR:=ADR(Merker) ,
    READ:=TRUE ,
    TMOUT:=t#1s );
IF NOT fbADRSREAD.BUSY THEN
    fbADRSREAD(READ:=FALSE);
END_IF
```

5.6 Local process image in delivery state

The process image of the BX/BCxx50 consists of an input, output and flag area. In addition, there are unallocated data without fixed address. They are created without specifying an address. For this type of variable 256/48 kB of memory are available on the BX/BCxx50. The maximum size of a variable or structure (array) is 16 kbyte. For the allocated data 2048 bytes of input data and 2048 bytes of output data are available. The BX/BCxx50 has 4 kB of memory for the allocated flag area. In the delivery state (default configuration) of the BX/BCxx50, fixed addresses are allocated for all connected Bus Terminals. The data for fieldbus communication start from address offset 1000_{dec}. The fieldbus data length depends on the fieldbus system used.

Inputs	Outputs
Bus Terminal %IB0 ...	Bus Terminal %QB0 ...
Fieldbus data (PLC variables) %IB1000 ...	Fieldbus data (PLC variables) %QB1000 ...
... %IB2047 maximal	... %QB2047 maximal

Differences in the protocols

The data for the higher-level master, Modbus or KS8000 protocol are stored in the BC8150 from address 1000. For Modbus the maximum data length is 512 bytes and does not have to be set within the range 1 to 126 words. For the KS8000 protocol the number of data from address 1000 is preset to 8 words or 16 bytes inputs and 8 words or 16 bytes outputs. The BC8150 will reject data telegrams that do not match this number exactly and issue an error. This data length can be parameterized via a function block in the BC8150.

If the Modbus protocol is used, data access to the BC8150 flag area is recommended. This area can be read and or written to without triggering the watchdog. The ADS protocol can access all data. The whole process image can be accessed via port 800, and the process image from address 1000 via port 300. If port address 800 is used, the data are only available for read access. If the serial ADS is used, it is recommended to access the flag area that can be reached via port 800 and IndexGroup 0x4020.

Addressing of the connected Bus Terminals

The default setting is for all the connected Bus Terminals to be assigned to the local process image. Mapping within the Bus Terminal Controller is carried out according to the following rule:
First all the complex Bus Terminals, in the sequence they are physically inserted, followed by the digital Bus Terminals which are filled to a byte. The default mapping of the complex Bus Terminals is:

- complete evaluation
- Intel format
- Word Alignment

Example structure

Bus Terminal Controller: 1 x BX/BCxx50
Position 1: 1 x KL1012
Position 2: 1 x KL1104
Position 3: 1 x KL2012
Position 4: 1 x KL2034
Position 5: 1 x KL1501
Position 6: 1 x KL3002
Position 7: 1 x KL4002
Position 8: 1 x KL6001
Position 9: 1 x KL9010

Process image

Bus Terminal	Position	Input image	Output image
KL1501	5	%IB0...%IB5	%QB0...%QB5
KL3002	6	%IB6...%IB13	%QB6...%QB13
KL4002	7	%IB14...%IB21	%QB14...%QB21
KL6001	8	%IB22...%IB29	%QB22...%QB29
KL1012	1	%IX30.0..30.1	-
KL1104	2	%IX30.1..30.5	-
KL2012	3	-	%QX30.0..30.1
KL2034	4	-	%QX30.2..30.5
KL9010	9	-	-

i **Show associated variables**

If you do not know the address of the Bus Terminals that you have assigned to the local PLC (BX/BCxx50):
Perform your hardware configuration in the System Manager. After you have entered all the Bus Terminals and PLC variables, click with the right mouse button on the BX/BCxx50 in the hardware tree, and select the menu item *Export variables information*.... A file is saved, and this file can be inserted in the System Manager under *Project - Import*. Now you will have the entry *TwinCAT import* under the global variables, and you will find here all the variables that you have assigned to the local PLC (BX/BCxx50).

5.7 Mapping the Bus Terminals

The precise assignment of the byte-oriented Bus Terminals may be found in the configuration guide for the particular bus terminal. This documentation can be found on the Internet at http://www.beckhoff.de.

Byte oriented Bus Terminals	Bit oriented Bus Terminals
KL15x1	KL10xx, KL11xx, KL12xx, KL17xx, KM1xxx
KL25xx	KL20xx, KL21xx, KL22xx, KL26xx, KL27xx, KM2xxx
KL3xxx	
KL4xxx	
KL5xxx	
KL6xxx	
KL7xxx	
KL8xxx	
	KL9110, KL9160, KL9210, KL9260

5.8 Local process image in the TwinCAT configuration

The TwinCAT configuration (TwinCAT CONFIG) enables free mapping between fieldbus, K-bus and PLC variables. Variables can be linked independent of their address via the System Manager.
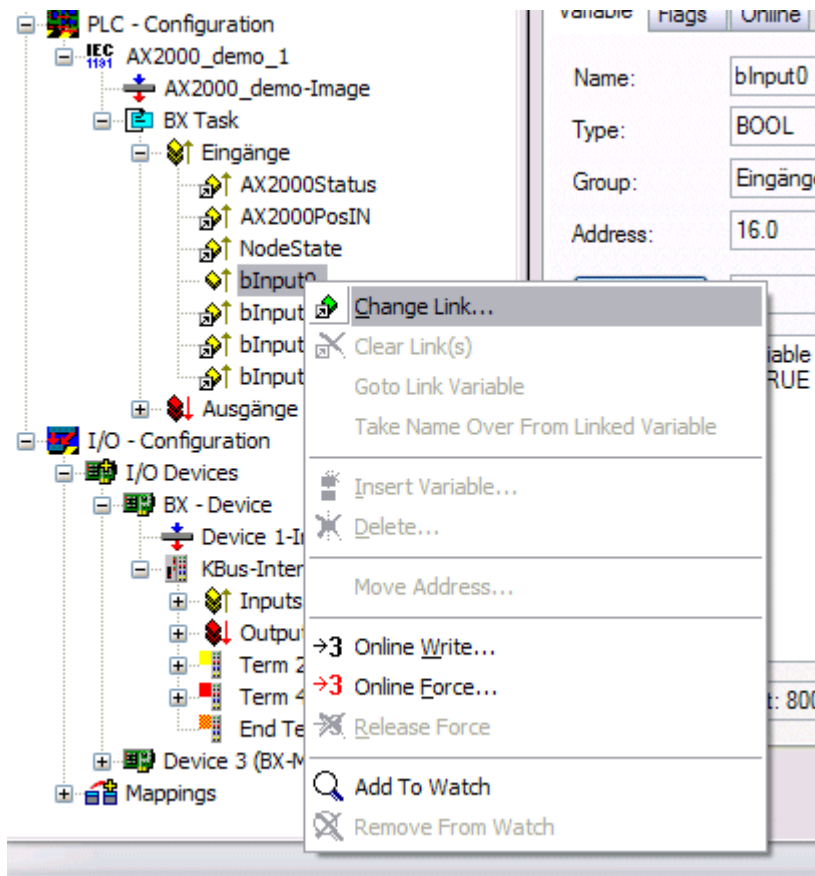
Fig. 46: Changing variable links
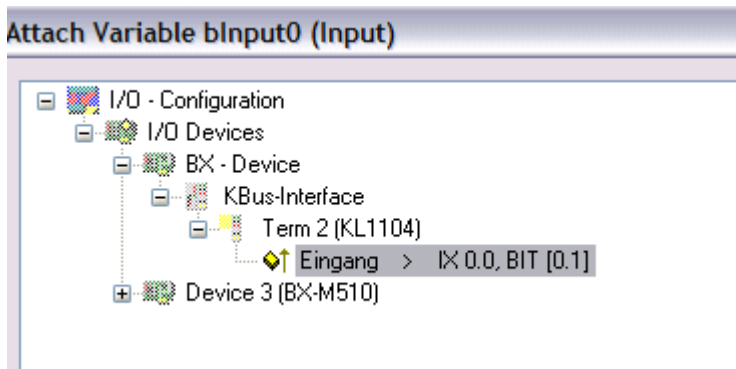
Fig. 47: Linking a variable with an input

In the default configuration all Bus Terminals are assigned fixed addresses. If a Bus Terminal is inserted, the whole address range may be shifted. The TwinCAT configuration enables allocated variables to be linked to a Bus Terminal, as required. This is parameterized in the System Manager, and the configuration is then downloaded to the Bus Terminal Controller (see TwinCAT configuration [▶ 28]). It is also possible to upload an existing TwinCAT configuration.

5.9 Creating a boot project

The following memory resources are available for generating the boot project

- approx. 250 kbyte flash on the Bus Terminal controllers of the BX series;
- approx. 48 kbyte flash on the Bus Terminal controllers of the BCxx50 series.

PLC Control

After logging into TwinCAT PLC Control, a boot project can be created.

- Opening a PLC project
- Selecting the target system (or selection the serial interface)
- Logging into the BX/BCxx50
- Creating a boot project (Online\Create boot project)

The PLC LED lights up green once a valid boot project is available on the BX/BCxx50.

In the Bus Terminal controllers of the BX series, the PLC LED flashes orange while boot project is created. The PLC LED lights up orange if no boot project is available on the BX.

Deleting a boot project

The boot project can be deleted from the Bus Terminal Controller. The following steps must be followed:

- Opening the project
- Logging into the Bus Terminal Controller
- Deleting the boot project (Online\Delete boot project)

The PLC LED lights up orange when the boot project is deleted.

ⓘ **Using the current project as boot project**

After an online change the old project is still shown as boot project. To use the current project (after the online change) as the boot project, the boot project has to be recreated.

Bypassing the start of the boot project*

With the Bus Terminal controllers of the BX series, starting of the boot project during booting can be prevented by pressing the Navi button. This does not delete the boot project. The project is reloaded when the Bus Terminal Controller is rebooted.

* from version 0.85

5.10 Communication between TwinCAT and BX/BCxx50

For transferring data from TwinCAT to the Bus Terminal Controller, it makes sense to organize the data in a structure. Please note the following to account for the differences in data management on the two systems.

- If two different data types are sent in sequence (e.g. byte and INT), the following variable is set to the next even address offset
- Boolean variables should never be allocated individually within a structure, since they would invariably occupy 1 byte. Boolean expressions should always be masked in a byte or word.

Example 1: A structure on the BX/BCxx50 and on the PC

Variable	BX/BCxx50 memory	PC memory (TwinCAT)
Byte	%..B0	%..B0
INT (1)	%..B2	%..B1
INT (2)	%..B4	%..B3

Due to the fact that another variable type (INT) follows the first byte, in the BX/BCxx50 it was assigned the next free even address. In order to achieve the same data structure on both systems, a dummy byte has to be inserted in the PC project (see example 2).

Example 2: A structure on the BX/BCxx50 and on the PC with the same memory allocation

Variable	BX/BCxx50 memory	PC memory (TwinCAT)
Byte	%..B0	%..B0
Byte (dummy)	%..B1 (not necessarily required, since the system deals with this itself if the variable does not exist)	%..B1
INT (1)	%..B2	%..B2
INT (2)	%..B4	%..B4

Data structure

```
Type PB_Data
STRUCT
    wVar_1:WORD;
    iValue_1:INT;
    iValue_2:INT;
    iValue_3:INT;
END_STRUCT
END_TYPE
```

Creating a variable structure

```
VAR_Global
    strData_Out AT %QB1000:PB_Data; (*PLC Variables *)
    bInput_01 AT %IX0.0:BOOL; (* Input from a terminal *)
END_VAR
```

Small programming example

```
strData_Out.wVar_1.0:=bInput_01;
```

> ℹ **Do not use real values in a mixed data structure**
>
> A mixed data structure should not contain real values. If this is nevertheless the case, the high and low words must be swapped in the BX/BCxx50 or in the TwinCAT master project. It is better to use an array of Real values or to transfer the Real values individually.

> ℹ **Larger fieldbus data blocks**
>
> You can transfer larger fieldbus data blocks, in order to have a reserve for your structure. Disadvantage: These reserves are then transferred with each fieldbus telegram, resulting in overload of the fieldbus communication.

5.11 Up- and downloading of programs

The Bus Terminal Controller has a memory for the source code. It can be used for storing the program, the task configuration, and the libraries. Should the memory be insufficient, the source code may be stored without task configuration and libraries. This takes up significant less memory space!

General settings

The timing of the source code download to the target system can be specified via Edit/Options. Open the options menu.
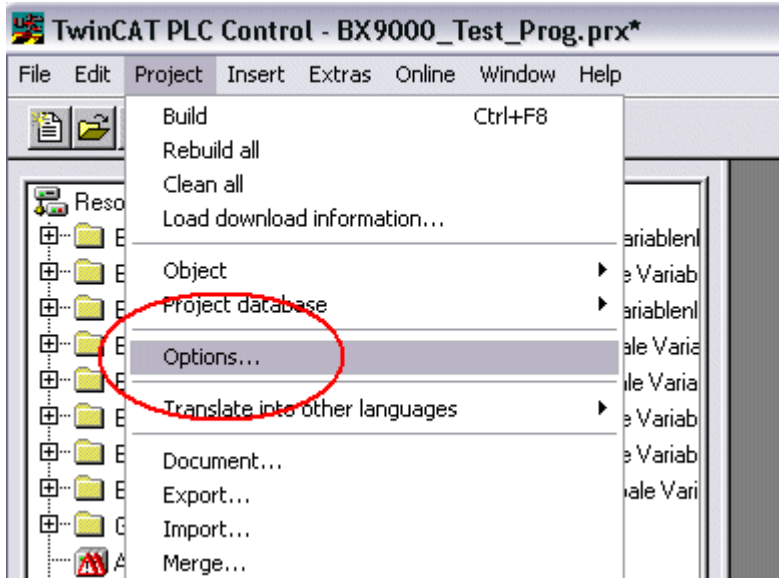
Fig. 48: Opening the options menu

Select Source Download.
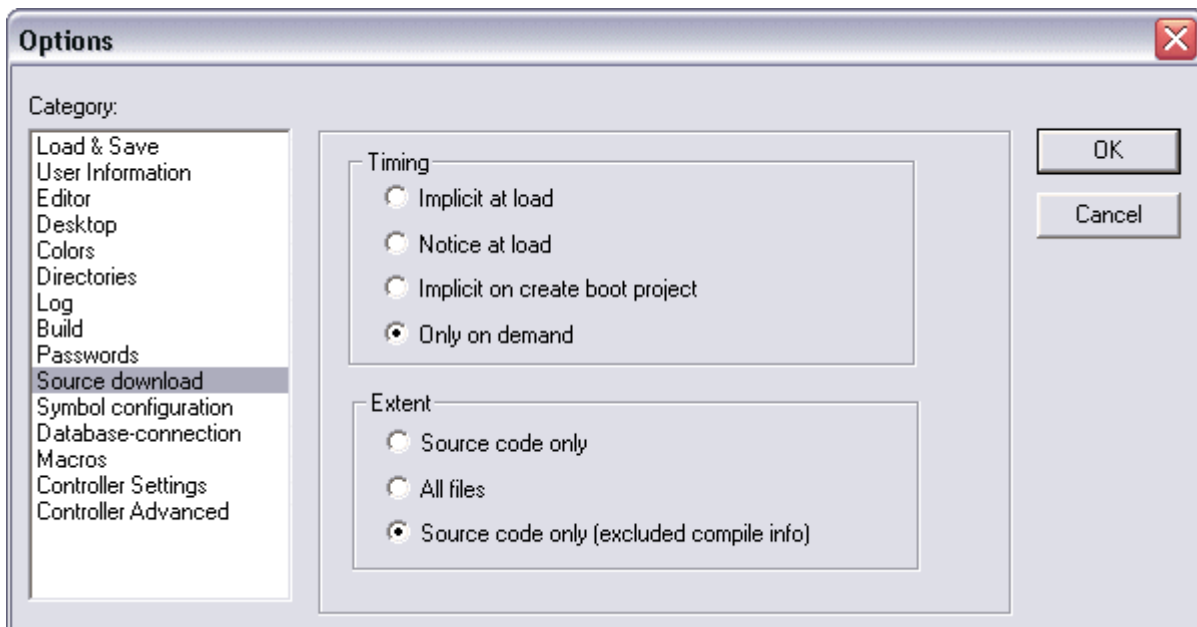
Fig. 49: Selecting Source Download

Here you can set which parts of the source code are to be downloaded to the Bus Terminal Controller, and when.

Source code only: the prx file with information on the online change is transferred. Login via online change is possible (the PLC does not stop).
All files: as *Source code only*, plus all required libraries.
Source code only (compile info excluded): only the prx file is transferred. Login is only possible when the PLC stops.

Which option you can use depends on the size of your projects.

Downloading a program

The source code can be transferred to the target system on request. This requires the user to be logged in with his program. Under Online/Source code download the program code can now be transferred to the Bus Terminal Controller.
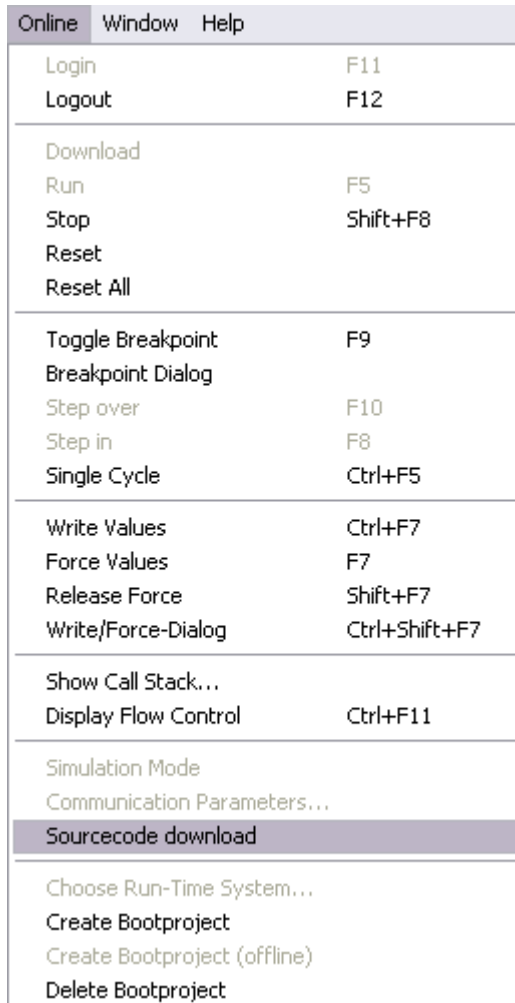
Fig. 50: Downloading the program code

After a short delay, a window will open that indicates the download progress.
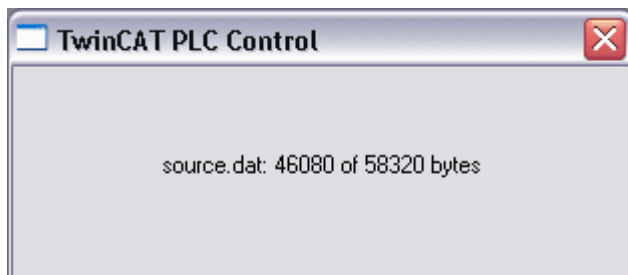
Fig. 51: Download progress

Uploading a program

For uploading the program code again, open a new file in PLC Control. Then click on the PLC button.
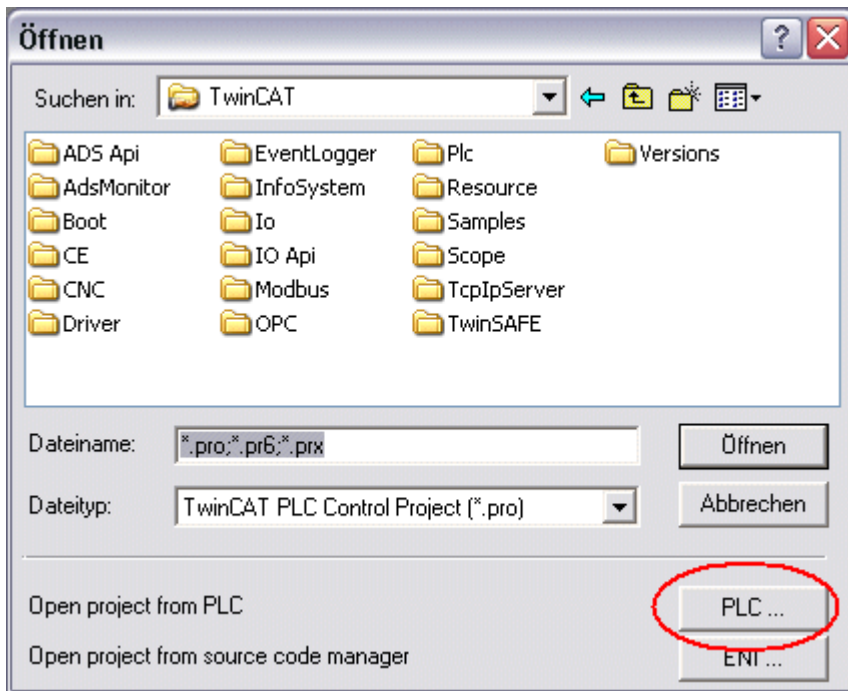
Fig. 52: Uploading a program

Select the data transfer route:

- *BCxx50 or BX via AMS*, if you are connected to the Bus Terminal Controller via the fieldbus, or
- *BCxx50 or BX via serial*, if you are connected to the Bus Terminal Controller via the serial interface.
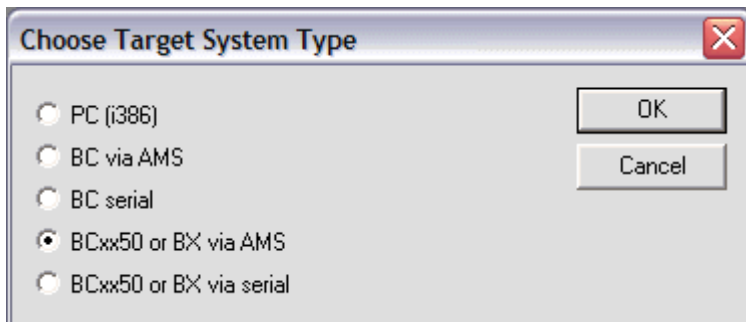
Fig. 53: Selecting the data transfer route

Then select the device and confirm with OK.
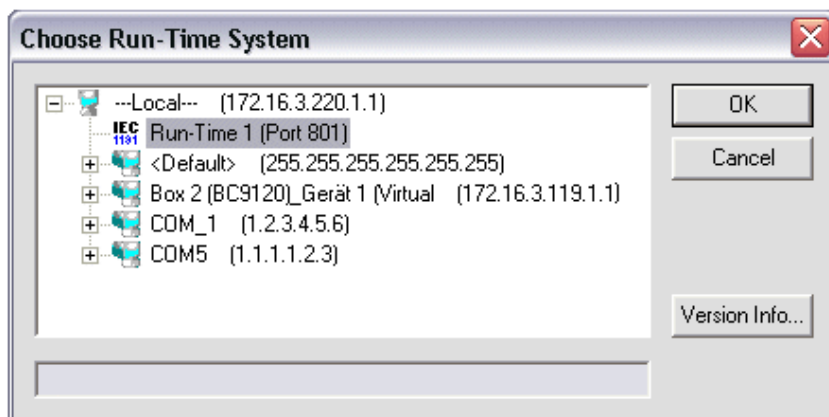
Fig. 54: Selecting the device

The source code will now be uploaded.

Password

You can protect your project with a password (in PLC Control Project/Options/Passwords).

5.12 Libraries

5.12.1 Libraries overview

The TwinCAT Automation Software offers various libraries for the BCxx50 series Bus Terminal Controllers (Bus Coupler with PLC functionality). See Beckhoff Information System.

Download

The libraries are also included in this documentation. To extract the libraries, left-click on the link and copy the libraries to directory TwinCAT\PLC\LIB.

Standard
(https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/3730726795.zip)

TcSystemBCxx50. TcSystemBCxx50 requires the TcBaseBCxx50 library.
(https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/3730728971.zip)

TcBaseBCxx50
(https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/3730731147.zip)

ChrAscBX.lbx
(https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/3730733323.zip)

> ℹ️ **Use the library that matches the firmware**
> The latest firmware requires the latest library. If you update your Bus Terminal Controller, please also update the libraries.
> Copy these libraries to the LIB directory, remove these libraries from your project and add them again.

TcSystemBCxx50

ADS	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
ADSREAD		B0	B0	B1	B0	-
ADSWRITE		B0	B0	B1	B0	-
ADSRDWRT		B0	B0	B1	B0	-
ADSWRTCTL		B0	B0	B1	B0	-
ADSRDSTATE		B0	B0	B1	B0	-
ADSRDDEVINFO		B0	B0	B1	B0	-

Bit Functions	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
CLEARBIT32		B0	B0	B1	B0	-
CSETBIT32		B0	B0	B1	B0	-
GETBIT32		B0	B0	B1	B0	-
SETBIT32		B0	B0	B1	B0	-

Controller	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
FB_BasicPID	-	B0	B0	B1	B0	-
-	-	-	-	-	-	-

File Access	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
FB_ReadFromFile		-	-	-	-	-
FB_WriteToFile		-	-	-	-	-
FB_ReadWriteFile		-	-	-	-	-

Memory Functions	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
MEMCMP		B0	B0	B1	B0	-
MEMCYP		B0	B0	B1	B0	-
MEMMOVE		B0	B0	B1	B0	-
MEMSET		B0	B0	B1	B0	-

NOVRAM Functions	Version	Firmware				
		BX3100	BX5100	BX5200	BX8000	-
-	-	-	-	-	-	-

SFC	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
AnalyzeExpression		-	-	-	-	-
AppendErrorString		-	-	-	-	-
SFCActionControl		-	-	-	-	-

System / Time / TBus	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
DRAND		B0	B0	B1	B0	-
SYSTEMTIME_TO_DT		B0	B0	B1	B0	-
DT_TO_SYSTEMTIME		B0	B0	B1	B0	-
GetSysTick		B0	B0	B1	B0	-
PresetSysTick		B0	B0	B1	B0	-
Reboot		B0	B0	B1	B0	-

Debug	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
F_ReadDebugTimer		B0	B0	B1	B0	-
F_StartDebugTimer		B0	B0	B1	B0	-

5.12.2   Seriell_Set data structure

The RS232-specific function blocks enable specific serial parameters to be set in the default configuration of the BC8150. It is therefore possible to set the length of the I/O data, the baud rate, and the transfer mode from the local PLC project. The function block is not required for the TwinCAT configuration, as in this case the settings are created via the System Manager.

BC8150 library (old version)
(https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/4059878411.zip)

BC8150 library (current version)
(https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/4059880587.zip)

example for BC8150 and KS8000 protocol
(https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/4059882763.zip)

example for BC8150 and ModbusRTU protocol
(https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/4059884939.zip)

BC8150_Settings


```
                    BC8150_SETTINGS
── bWriteSet : BOOL                      bBusy : BOOL ──
── bReadSet : BOOL                       bError : BOOL ──
── strBC8150_Set : Seriell_Set          iErrorID : DINT ──
                              strBC8150 : Seriell_Setting ──
                            bReboot_Necessary : BOOL ──
```

Fig. 55: Function block BC8150_SETTINGS

FUNCTION_BLOCK BC8150_Settings

VAR_INPUT

```
bWriteSet        :BOOL;
bReadSet         :BOOL;
strBC8150_Set    :Seriell_Set;
```

Key

bWriteSet: Rising edge starts the block and reads/writes the parameters. If these have changed, the new parameters are written and the flag bReboot_Necessary is set. The Bus Coupler must then be restarted. The user is notified if/when the bReboot_Necessary flag is set.
bReadSet: Positive edge reads the set RS232 parameters.
str*BC8150_Set* : Data structure for the RS232 interface.

VAR_OUTPUT

```
bBusy              :BOOL;
bError             :BOOL;
iErrorId           :WORD;
strBC8150          :Seriell_Setting;
bReboot_Necessary  :BOOL;
```

Key

bBusy: The block is active as long it is TRUE.
bError: Error bit.
iErrorId: Error number.
str*BC8150*: Output of type, firmware, and RS232 parameters.
bReboot_Necessary: If this bit is TRUE after the bBusy is set to FALSE, then the BC8150 must be restarted so that the RS232 settings are saved.

Seriell_Set data structure

The settings for the RS232 interfaces are transferred with the following data structure.

```
TYPE ST_COMConfig:
STRUCT
   iInputData          :INT; (* BC8150: 1..255 *)
   iOutputData         :INT; (* BC8150: 1..255 *)
   Modbus_Mode_BC8150  :E_BC8150_Mode; (* _8data_no_parity, _8data_with_parity, _8data_even_parity,
_8data_odd_parity *)
   iWatchdog           :INT; (* 1000 ms Watchdog für die Modbus und KS8000 Kommunikation , 0 ms Watc
hdog disabled*)
   Modbus_Baudrate     :ENUM;(* _1k2Baud, _2k4Baud, _4k8Baud, _9k6kBaud, _19k2Baud, _38k4Baud *)
END_STRUCT
END_TYPE
```

Key

iInputData: Number of input data (from the point of view of the BC). Size in words for the KS8000 protocol (address 1-39)
iOutputData: Number of output data (from the point of view of the BX). Size in words for the KS8000 protocol (address 1-39)
Modbus_Mode_BC8150: RS232 settings for the ModbusRTU protocol (address 40-59)
iWatchdog: Watchdog for KS8000 and Modbus protocol
Modbus_Baudrate: Baud rate for the ModbusRTU protocol (address 40-59)

iErrorId

Return parameter iErrorId	Meaning
0	no Error
-1	Wrong BC Typ
-2	Wrong firmware version
0x1000_xxxx	xxxx ADS Error Code (see ADS Return Code)

5.12.3   TcBaseBCxx50

5.12.3.1   System task information

```
VAR_GLOBAL
    SystemTaskInfo  :  SYSTEMTASKINFOTYPE;
END_VAR
```

System flags are implicitly declared variables. Using the Input Assistant, a variable SystemTaskInfoArr can be found under system variables. This variable is a field with four structures of type SYTEMTASKINFOTYPE [▶ 61]. The structure definition can be found in the system library. The index in this field is the task ID.

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.9.0	BCxx50, BC9x20 Controller	TcBaseBCxx50.lbx

5.12.3.2 System Task Info Type

```
TYPE SYSTEMTASKINFOTYPE
STRUCT
    active           :        BOOL;
    taskName         :        STRING(16);
    firstCycle       :        BOOL;
    cycleTimeExceeded :       BOOL;
    cycleTime        :        UDINT;
    lastExecTime     :        UDINT;
    priority         :        BYTE;
    cycleCount       :        UDINT;
END_STRUCT
END_TYPE
```

Key

active: This variable indicates whether the task is active.

taskName: the task name.

firstCycle: During the first PLC task cycle, this variable has the value: TRUE.

cycleTimeExceeded: this variable indicates whether the set task cycle time was exceeded.

cycleTime: set task cycle time in multiples of 100 ns.

lastExecTime: cycle time required for the last cycle in multiples of 100 ns.

priority: set task priority.

cycleCount: cycle counter.

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.9.0	BCxx50, BC9x20 Controller	TcBaseBCxx50.lbx

5.12.3.3 System info

```
VAR_GLOBAL
    SystemInfo    : SYSTEMINFOTYPE;
END_VAR
```

System flags are implicitly declared variables. Using the Input Assistant, a variable Systeminfo can be found under system variables. The type SYSTEMINFOTYPE [▶ 61]is declared in the system library. For accessing the variable, the system library has to be integrated in the project.

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.9.0	BCxx50, BC9x20 Controller	TcBaseBCxx50.lbx

5.12.3.4 System information type

```
TYPE SYSTEMINFOTYPE
STRUCT
    runTimeNo        :        BYTE;
    projectName      :        STRING(32);
    numberOfTasks    :        BYTE;
    onlineChangeCount :       UINT;
    bootDataFlags    :        BYTE;
    systemStateFlags :        WORD;
END_STRUCT
END_TYPE
```

Key

runTimeNo: indicates the number of the runtime system (1).

projectName: project name as STRING.

numberOfTasks: number of tasks contained in the runtime system (max. 1).

onlineChangeCount: number of online changes since the last complete download.

bootDataFlags: Reserved

systemStateFlags: Reserved.

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.9.0	BCxx50, BC9x20 Controller	TcBaseBCxx50.lbx

5.12.3.5 ADS

5.12.3.5.1 Local ADS Port Numbers

Port number	Description
100 [▶ 62] dec	Reading and writing of registers and tables from the coupler and the complex Bus Terminals
801 [▶ 62] dec	Local PLC process image

5.12.3.5.2 ADS services

Local process image task 1 port 800 or 801

Data can be read from and written to the local process image. If it is necessary for outputs to be written, it is important to ensure that they are not used by the local PLC, because the local controller will overwrite these values. The data are not associated with a watchdog, and therefore must not be used for outputs that would have to be switched off in the event of a fault.

Index group	Meaning	Index offset (value range)
0xF020	Inputs	0...2047
0xF030	Outputs	0...2047
0x4020	Flags	0...4095

ADS services

AdsServerAdsState

Data type (read only)	Meaning
String	Start - the local PLC is running Start - the local PLC is stopped

AdsServerDeviceState

Data type (read only)	Meaning
INT	0: Start - the local PLC is running 1: Start - the local PLC is stopped

AdsServerType

Data type (read only)	Meaning
String	BX PLC Server

Register Port 100

On the Bus Terminal Controllers of the BX series, and on the BCxx50, the ADS port number for register communication is fixed at 100.

Index Group	Index offset (value range)		Meaning
	Hi-Word	Lo-Word	
0 [READ ONLY]	0...127	0..255	Registers in the Bus Coupler Hi-Word, table number of the Bus Coupler Lo-Word, register number of the table
1...255	0...3	1-255	Register of the Bus Terminals Hi-Word, channel number Lo-Word, register number of the Bus Terminal

> **i** **Minimum timeout**
>
> When reading the register, the time out of the ADS block has to be set to a time longer than 1 second.

> **i** **Setting the password**
>
> When writing to the registers, the password has to be set (see the documentation for the particular Bus Terminal).

5.13   Program transfer

5.13.1   Program transfer via the serial interface

Every Bus Terminal Controller can be programmed via the PC's RS232 interface.

Select the serial interface in TwinCAT PLC Control.
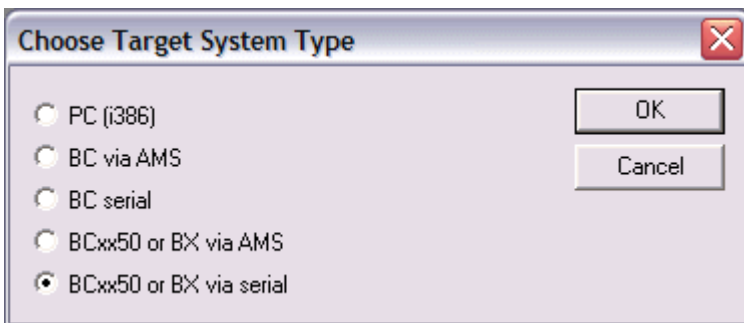
Fig. 56: *Selecting the data transfer route - serial interface*

The settings for the serial interface, port number, baud rate etc. are found under Online/Communication parameters in PLC Control.

The Bus Terminal Controller requires the following setting:

- Baud Rate: 9600/19200/38400/57600 baud (automatic baud rate detection)
- Stop bits: 1
- Parity: Straight line
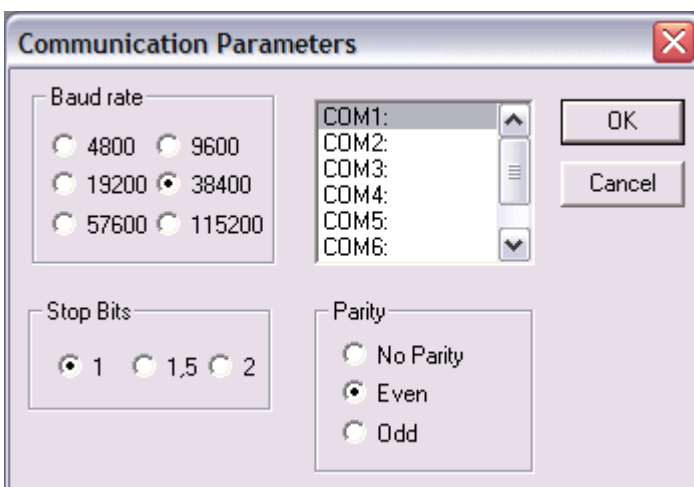
Fig. 57: Parameterization of the serial interface

Program transfer via the serial interface and ADS

The Bus Terminal Controller can be programmed via the PC's RS232 interface. Before you can work with the Bus Terminal Controller, TwinCAT must be notified of it (see serial ADS [▶ 36]).

Select the ADS connection in TwinCAT PLC Control.
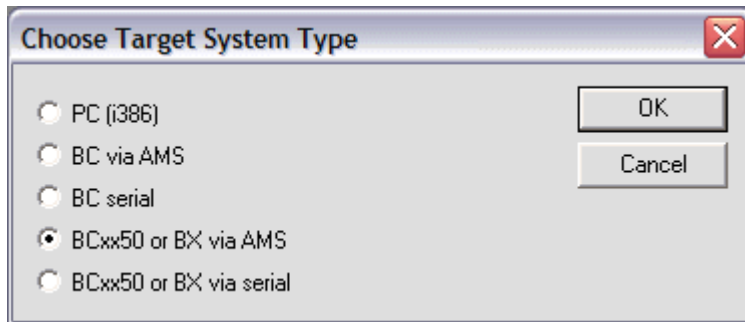
Fig. 58: *Selecting the data transfer route - AMS*

PLC Control can be accessed via *Online/Communication Parameters...*.
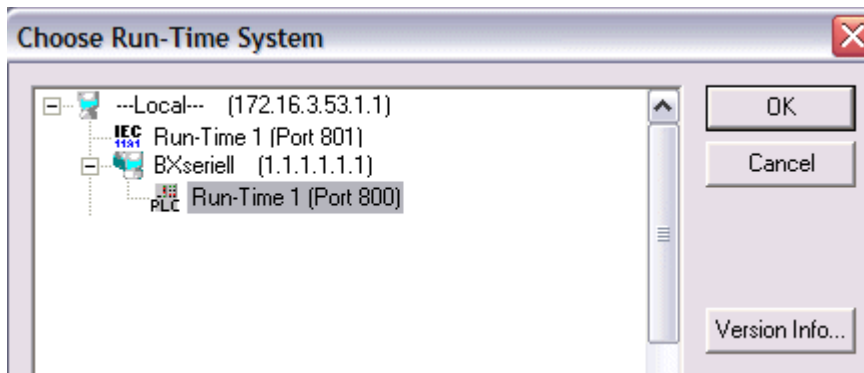
Fig. 59: Selecting the device

5.14 Process image

5.14.1 Modbus Process Image

Default config (without TwinCAT System Manager):

The data of the Modbus slave interface are stored in the allocated section of the BC8x50.

The input data from the master perspective are stored in the controller output section from address %QB1000 (Modbus address 30001, depending on master).

The output data from the master perspective are stored in the controller input section from address %IB1000 (Modbus address 42049 - depending on master).

The maximum data length is 512 bytes input and 512 bytes output.

Recommendation

We recommend limiting Modbus access to the flag area of the controller, master address 3 16385_{dec} or 4 16385_{dec} (3 4001_{hex} or 4 4001_{hex}, see Modbus interface [▶ 67]). The flag area has a size of 4 kB and is not watchdog-triggered. The flag area is available for reading or writing.

5.14.2   KS8000 Process Image

Default config (without TwinCAT System Manager):

The data of the KS8000 slave interface are stored in the allocated section of the BC8x50.

The input data from the master perspective are stored in the controller output section from address %QB1000 (address offset 0 from the master perspective).

The output data from the master perspective are stored in the controller input section from address %IB1000 (address offset 0 from the master perspective).

The maximum data length is 512 bytes input and 512 bytes output.

6 RS232 - Interface

6.1 Protocol

6.1.1 Modbus

6.1.1.1 Modbus protocol

RTU

In RTU mode the protocol starts with a pause of 3.5 characters, and finishes in the same way (illustrated in the diagram with T1-T2-T3-T4).
The characters permitted for transmission in all fields are hexadecimal 0... 9, A..., F.

Start	Slave address	Function	Data	CRC	END
t1-t2-t3-t4	1 byte	1 byte	n Bytes	2 bytes	t1-t2-t3-t4

ASCII

In ASCII mode the telegram starts with a colon (":", character 0x3A) and ends with a *carriage return line feed* ("CRLF", characters 0x0D and 0x0A).
The characters transferred are represented in the ASCII code.

Start	Slave address	Function	Data	LRC	END
1 byte 0x3A	2 bytes	2 bytes	n Bytes	2 bytes	2 bytes 0x0D, 0x0A

6.1.1.2 Data Frame

Description	ModbusRTU (address 40-59) DEFAULT	ModbusASCII (address 1-39) DEFAULT
Data bits	8	8
Stop bits	1	1
Baud rate	9600 baud	38400 baud (automatic detection)
Parity	none	even

6.1.1.3   Modbus interface

Address		Description		
0x0000 0x00FF		Process data interfaces Inputs		
0x0800 0x08FF		Process data interfaces Outputs		
0x1000 0x1006	Read only	Bus Coupler Fieldbus Box ID		
0x100A		2 bytes PLC interface		
0x100B		Bus terminal diagnosis		
0x100C		Bus Coupler status		
0x1010		Process image length in bits, analog outputs (without PLC variables)		
0x1011		Process image length in bits, analog inputs (without PLC variables)		
0x1012		Process image length in bits, digital outputs		
0x1013		Process image length in bits, digital inputs		
0x1020		Watchdog, current time in [ms]		
0x110A	Read/Write	2 bytes PLC interface		
0x110B		Bus terminal diagnosis		
0x1120		Watchdog, pre-defined time in [ms] (Default value: 1000)		
0x1121		Watchdog reset register		
0x1122		Type of watchdog	1_{bin}	Telegram watchdog (default)
			0_{bin}	Write telegram watchdog
0x4000 0x40FF*		Flags area (%MB..)*		

* only for Bus Terminal Controllers (BC7300, BC8x50, IL230x-C730)

Watchdog

The watchdog is active under the factory settings. After the first write telegram the watchdog timer is initiated, and is triggered each time a telegram is received from this device. A second approach, which represents a more sensitive condition for the watchdog, is for the watchdog only to be re-triggered after each write telegram. To do this, write a zero into register 0x1122 (default value "1").
The watchdog can be deactivated by writing a zero to offset 0x1120. The watchdog register can only be written if the watchdog is not active. The data in this register is retained.

Watchdog register

If the watchdog timer on your slave has elapsed it can be reset by writing twice to register 0x1121. The following must be written to the register: 0xBECF 0xAFFE. This can be done either with function 6 or with function 16.

Bus Coupler status

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FB	-	-	-	-	-	-	-	-	-	-	-	-	-	CNF	KB

FB: Fieldbus error, watchdog expired
KB: Bus Terminal error/IP-Link error
CNF: Bus Coupler configuration error

2 byte PLC interface*

Registers in the complex terminals and Bus Terminal Controller registers can be both read and written using the 2 byte PLC interface. The complex terminal registers are described in the associated terminal documentation. The Bus Coupler registers can be used, for example, to read terminal bus diagnostics data,

the terminal composition or the cycle times, and the programmed configuration can be written. It is also possible for a manual K-bus reset to be carried out. The 2-byte PLC interface requires two bytes each of input and output data. They are handled using a special protocol. A description of the 2 byte PLC interface, the registers available in the Bus Couplers and of function blocks for various PLC systems that support the 2 byte PLC interface can be supplied on request.

* not implemented on the BC8150

2 byte diagnostic interface

The terminals' error messages can be sent over the 2-byte diagnostic interface. K-bus diagnostics must however be activated for this purpose. The 2-byte diagnostic interface occupies two bytes each of input and output data. A special protocol is processed via these two bytes. A description of the 2 byte-diagnostic interface can be supplied on request.

6.1.1.4 Modbus error response

When the user sends the slave either a request or information that the coupler does not understand, the slave responds with an error report. This answer contains the function and the error code. 0x80 is added to the value returned by the function.

Code	Name	Meaning
1	ILLEGAL FUNCTION	Modbus function not implemented
2	ILLEGAL DATA ADDRESS	Invalid address or length
3	ILLEGAL DATA VALUE	Invalid parameters - Diagnostics functions - Wrong register
4	SLAVE DEVICE ERROR	Watchdog or K-bus error

6.1.1.5 Functions

6.1.1.5.1 Modbus functions in the BC8150

In the Modbus protocol, the functions determine whether data is to be read or written, and what kind of data is involved.

Function	Code	Description
Read holding register [▶ 68]	3	Read analog outputs and inputs / GPR
Read input register [▶ 70]	4	Read analog inputs / GPR
Preset single register [▶ 71]	6	Write one analog output / GPR
Diagnosis [▶ 72]	8	Diagnosis
Preset multiple register [▶ 74]	16	Write a number of analog outputs / GPRs
Read / write registers [▶ 75]	23	Write and read a number of process data outputs / GPRs

GPR (General Preset Register) - register structure of the Modbus interface (see Appendix)

6.1.1.5.2 Function 3: READ HOLDING REGISTERS

The READ HOLDING REGISTERS function can be used to read the input and output words and the registers. The inputs have offsets 0 - 0xFF while the outputs have offsets 0x800 - 0x8FF.

The first two analog outputs are read in this example. The analog outputs begin at offset 0x800 (hex). The length indicates the number of channels to be read.

Query

Byte Name	Example			
Start - frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	3	0x03	"03"	0x30, 0x33
Start address high	8	0x08	"08"	0x30, 0x38
Start address low	0	0x00	"00"	0x30, 0x30
Count high	0	0x00	"00"	0x30, 0x30
Count low	2	0x02	"02"	0x30, 0x32
Error check, LRC/ CRC		0xC6, 0xC1	"E8"	0x45, 0x38
End - frame		t1-t2-t3	CRLF	0x0D, 0x0A

The Fieldbus Box Coupler Box responds with byte count 4, i.e. 4 bytes of data are returned. The query was for 2 analog channels, and these are now distributed over 2 words. In the analog output process image, the first channel has the value 0x3FFF, while the second channel has the value 0x0.

Response

Byte Name	Example			
Start - frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	3	0x03	"03"	0x30, 0x33
Byte Count	4	0x04	"04"	0x30, 0x34
Data 1 high byte	63	0x3F	"3F"	0x33, 0x46
Data 1 low byte	255	0xFF	"FF"	0x46, 0x46
Data 2 high byte	0	0x00	"00"	0x30, 0x30
Data 2 low byte	0	0x00	"00"	0x30, 0x30
Error check, LRC/ CRC		0xC6, 0xC1	"B0"	0x42, 0x30
End - frame		t1-t2-t3	CRLF	0x0D, 0x0A

6.1.1.5.3 Function 4: READ INPUT REGISTERS

The READ INPUT REGISTERS function reads the analog inputs.

In this example the first two analog inputs of slave number 11 are read. The analog outputs start at an offset of 0x0000. The length indicates the number of words to be read. A KL3002 has 2 words of input data, which is why the value to be entered in "Count low" is two.

Query

Byte Name	Modbus RTU		Modbus ASCII	
Start - frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	4	0x04	"04"	0x30, 0x34
Start address high	0	0x00	"00"	0x30, 0x30
Start address low	0	0x00	"00"	0x30, 0x30
Count high	0	0x00	"00"	0x30, 0x30
Count low	2	0x02	"02"	0x30, 0x32
Error check, LRC/ CRC		0x71, 0x61	"EF"	0x45, 0x46
End - frame		t1-t2-t3	CRLF	0x0D, 0x0A

The Fieldbus Box Coupler Box responds with byte count 4, i.e. 4 bytes of data are returned. The query was for 2 analog channels, and these will now be distributed over 2 words. In the analog input process image, the first channel has the value 0x0038, while the second channel has the value 0x3F1B.

Response

Byte Name	Modbus RTU		Modbus ASCII	
Start - frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	4	0x04	"04"	0x30, 0x34
Byte Count	4	0x04	"04"	0x30, 0x34
Data 1 high byte	0	0x00	"00"	0x30, 0x30
Data 1 low byte	56	0x38	"38"	0x33, 0x38
Data 2 high byte	63	0x3F	"3F"	0x33, 0x46
Data 2 low byte	11	0x0B	"0B"	0x30, 0x42
Error check, LRC/ CRC		0x80, 0x7E	"6A"	0x36, 0x41
End - frame		t1-t2-t3	CRLF	0x0D, 0x0A

6.1.1.5.4   Function 6: writing an analog output

Function 6 can be used to access the output process image and the interface.
The first analog output is written with function 6. The analog outputs start at an offset of 0x0800. Here again the offset always describes a word. This means offset 0x0803 refers to the fourth word in the output process image.

Query

Byte Name	Modbus RTU		Modbus ASCII	
Start - frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	6	0x06	"06"	0x30, 0x36
Start address high	8	0x08	"08"	0x30, 0x36
Start address low	0	0x00	"00"	0x30, 0x30
Data high	63	0x3F	"3F"	0x33, 0x46
Data low	255	0xFF	"FF"	0x46, 0x46
Error check, LRC/ CRC		0xDA, 0xB0	"A9"	0x41, 0x39
End - frame		t1-t2-t3	CRLF	0x0D, 0x0A

The Fieldbus Coupler Box responds with the same telegram and confirmation of the received data.

Response

Byte Name	Modbus RTU		Modbus ASCII	
Start - frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	6	0x06	"06"	0x30, 0x36
Start address high	8	0x08	"08"	0x30, 0x36
Start address low	0	0x00	"00"	0x30, 0x30
Data high	63	0x3F	"3F"	0x33, 0x46
Data low	255	0xFF	"FF"	0x46, 0x46
Error check, LRC/ CRC		0xDA, 0xB0	"A9"	0x41, 0x39
End - frame		t1-t2-t3	CRLF	0x0D, 0x0A

6.1.1.5.5   Function 8: Diagnosis

The diagnosis function provides a series of tests for checking the communication system between the master and the slave and for examining a variety of internal error states within the slave. A broadcast telegram is not supported.
The function in the query uses a subfunction code field of two bytes to define the type of test to be carried out. The slave outputs the function code and the subfunction code in a response.
The diagnostics queries use a two-byte data field to send diagnostics data or control information to the slave.

Query

Byte Name	Modbus RTU		Modbus ASCII	
Start - frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	8	0x08	"08"	0x30, 0x38
Subfunction high	0	0x00	"00"	0x30, 0x30
Subfunction low	0	0x00	"00"	0x30, 0x30
Data high	2	0x02	"02"	0x30, 0x32
Data low	3	0x03	"03"	0x30, 0x33
Error check, LRC/ CRC		0xA1, 0xC0	"E8"	0x45, 0x38
End - frame		t1-t2-t3	CRLF	0x0D, 0x0A

Response

Byte Name	Modbus RTU		Modbus ASCII	
Start - frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	8	0x08	"08"	0x30, 0x38
Subfunction high	0	0x00	"00"	0x30, 0x30
Subfunction low	0	0x00	"00"	0x30, 0x30
Data high	2	0x02	"02"	0x30, 0x32
Data low	3	0x03	"03"	0x30, 0x33
Error check, LRC/ CRC		0xA1, 0xC0	"E8"	0x45, 0x38
End - frame		t1-t2-t3	CRLF	0x0D, 0x0A

Mirroring a request (subfunction 0)

Subfunction 0 causes the data that is sent to the slave by the master to be returned.

Coupler Reset (Subfunction 1)

The Bus Coupler/Fieldbus Box is reinitialized with subfunction 1, error counters are reset, and the Bus Coupler/Fieldbus Box carries out a self-test. During the time, in which the Bus Coupler/Fieldbus Box is reset, no telegrams are received or sent.

● **Response with subfunction 1**
i Before the Bus Terminal Controller restarts, it sends a response with subfunction 1.

● **Subfunctions 0 and 1**
i The BC8150 has subfunctions 0 and 1 implemented. All other subfunctions return "0".

Subfunction	Data field (query)	Data field (response)
00 01	00 00	00 00

Delete all counter contents (Subfunction 10)

If this subfunction is called, the Bus Coupler/Fieldbus Box deletes all error counters.

Subfunction	Data field (query)	Data field (response)
00 0B	00 00	Echo query data

Bus communication response counter (subfunction 11)

Returns the number of communication answers.

Subfunction	Data field (query)	Data field (response)
00 0C	00 00	Counter value

CRC error counter (subfunction 11)

Indicates the number of detected CRC errors

Subfunction	Data field (query)	Data field (response)
00 0D	00 00	Counter value

Error response counter (subfunction 13)

This counter indicates the number of error response telegrams, which the Bus Coupler/Fieldbus Box has sent.

Subfunction	Data field (query)	Data field (response)
00 0D	00 00	Counter value

The following functions contain the counter states for various units. This means that the Modbus telegrams can be distinguished through units, for example when the Bus Coupler/Fieldbus Box is accessed from different masters.

Slave responses (subfunction 14)

Contains the number of answers that the slave has sent.

Subfunction	Data field (query)	Data field (response)
00 0E	00 00	Counter value

Number of unanswered telegrams (subfunction 15)

Contains the number of answers that the slave has not sent.

Subfunction	Data field (query)	Data field (response)
00 0F	00 00	Counter value

Number of error responses (subfunction 16)

Contains the number of error answers that the slave has sent.

Subfunction	Data field (query)	Data field (response)
00 10	00 00	Counter value

6.1.1.5.6   Function 16: Writing a number of analog outputs

Function 16 can be used to write a number of analog outputs. The first 2 analog output words are written in this example. The analog outputs start at an offset of 0x0800. Here the offset always describes a word. Offset 0x0003 writes to the fourth word in the output process image. The length indicates the number of words, and the *Byte* count is formed from the combination of all the bytes that are to be written.
Example: 4 words - correspond to byte count 8
The data bytes contain the values for the analog outputs. In this example, two words are to be written. The first word is to receive the value 0x7FFF, and the second word is to receive the value 0x3FFF.

Query

Byte Name	Modbus RTU		Modbus ASCII	
Start - frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	16	0x10	"10"	0x31, 0x30
Start address high	8	0x08	"08"	0x30, 0x38
Start address low	0	0x00	"00"	0x30, 0x30
Length high	0	0x00	"00"	0x30, 0x30
Length low	2	0x02	"02"	0x30, 0x32
Byte Count	4	0x04	"04"	0x30, 0x34
Date 1 byte 1	127	0x7F	"7F"	0x37, 0x46
Date 1 byte 2	255	0xFF	"FF"	0x46, 0x46
Date 2 byte 1	63	0x3F	"3F"	0x33, 0x46
Date 2 byte 2	255	0xFF	"FF"	0x46, 0x46
Error check, LRC/ CRC		0xCD, 0xE3	"1B"	0x31, 0x42
End - frame		t1-t2-t3	CRLF	0x0D, 0x0A

Response

The Bus Coupler/Fieldbus Box responds with the start address and the length of the sent words.

Byte Name	Modbus RTU		Modbus ASCII	
Start - frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	16	0x10	"10"	0x31, 0x30
Start address high	8	0x08	"08"	0x30, 0x38
Start address low	0	0x00	"00"	0x30, 0x30
Length high	0	0x00	"00"	0x30, 0x30
Length low	2	0x02	"02"	0x30, 0x32
Error check, LRC/ CRC		0x43, 0x02	"DB"	0x44, 0x42
End - frame		t1-t2-t3	CRLF	0x0D, 0x0A

6.1.1.5.7 Function 23: Writing and reading of several analog value outputs or inputs

A number of analog outputs can be written and a number of analog inputs read with one telegram using function 23. In this example the first 2 analog output words are written, and the first two analog inputs are read. The analog outputs start at offset 0x0800, while the inputs start at offset 0x0000. Here the offset always describes a word. Offset 0x0003 writes to the fourth word in the output process image. The length indicates the number of words, and the *Byte* count is formed from the combination of all the bytes that are to be written.
Example: 4 words - correspond to a byte count of 8
The data bytes contain the values for the analog outputs. In this example, two words are to be written. The first word is to receive the value 0x3FFF, and the second word is to receive the value 0x7FFF.

Query

Byte Name	Modbus RTU		Modbus ASCII	
Start - frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	23	0x17	"17"	0x31, 0x17
Read start address high	0	0x00	"00"	0x30, 0x30
Read start address low	0	0x00	"00"	0x30, 0x30
Read length high	0	0x00	"00"	0x30, 0x30
Read length low	2	0x02	"02"	0x30, 0x32
Write start address high	8	0x08	"08"	0x30, 0x38
Write start address low	0	0x00	"00"	0x30, 0x30
Write length high	0	0x00	"00"	0x30, 0x30
Write length low	2	0x02	"02"	0x30, 0x32
Byte Count	4	0x04	"04"	0x30, 0x34
Data 1 high	63	0x3F	"3F"	0x33, 0x46
Data 1 low	255	0xFF	"FF"	0x46, 0x46
Data 2 high	127	0x7F	"7F"	0x37, 0x46
Data 2 low	255	0xFF	"FF"	0x46, 0x46
Error check, LRC/CRC		0x76, 0xD3	"12"	0x31, 0x32
End - frame		t1-t2-t3	CRLF	0x0D, 0x0A

Response

The Bus Coupler/Fieldbus Box responds with the start address and the length of the transferred bytes in the byte count. The data information follows. In this example the first word contains 0x0038 (hex) while the second word contains 0x3F0B.

Byte Name	Modbus RTU		Modbus ASCII	
Start - frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	23	0x17	"17"	0x31, 0x17
Byte Count	4	0x04	"04"	0x30, 0x34
Data 1 high	0	0x00	"00"	0x30, 0x30
Data 1 low	56	0x38	"38"	0x33, 0x38
Data 2 high	63	0x3F	"3F"	0x33, 0x46
Data 2 low	11	0x0B	"0B"	0x30, 0x42
Error check, LRC/CRC		0xF8, 0xA7	"E3"	0x45, 0x33
End - frame		t1-t2-t3	CRLF	0x0D, 0x0A

6.1.2 Protocol Overview

BC8150 RS232

Protocols	Description	BC8150 settings
Serial ADS	Protocol for program download and data exchange	Address 1 up to 39, baud rate 38400, 8,e,1
KS8000 protocol	Protocol for exchanging data	Address 1 up to 39, baud rate 38400, 8,e,1
KS2000 protocol	Protocol for configuring Bus Terminals via the KS2000 configuration software	Address 1 up to 39, baud rate 38400, 8,e,1
TwinCAT PLC (BCxx50 or BX serial)	TwinCAT PLC, programming software for the program download	Address 1 up to 39, baud rate 38400, 8,e,1
Modbus ASCII	Open protocol for data exchange, Modbus ASCII SLAVE	Address 1 up to 39, baud rate 38400, 8,e,1
Modbus RTU	Open protocol for data exchange, Modbus RTU SLAVE	Address 40 up to 59, baud rate 9600, 8,n,1

BC8050 RS485

Protocols	Description	BC8150 settings
Serial ADS (only via the configuration interface)	Protocol for program download and configuration	Address 1 up to 39, baud rate 38400, 8,e,1
KS8000 protocol	Protocol for exchanging data	Address 1 up to 39, baud rate 38400, 8,e,1
KS2000 protocol	Protocol for configuring Bus Terminals via the KS2000 configuration software	Address 1 up to 39, baud rate 38400, 8,e,1
TwinCAT PLC (BCxx50 or BX serial)	TwinCAT PLC, programming software for the program download	Address 1 up to 39, baud rate 38400, 8,e,1
Modbus ASCII	Open protocol for data exchange, Modbus ASCII SLAVE	Address 1 up to 39, baud rate 38400, 8,e,1
Modbus RTU	Open protocol for data exchange, Modbus RTU SLAVE	Address 40 up to 59, baud rate 9600, 8,n,1

*

6.1.3 KS8000 protocol

6.1.3.1 Protocol Description

Transfer protocol

A simple transfer protocol is used for data communication with the Bus Coupler/Fieldbus Box. Process data exchange with the BK8x00/IPxxxx-B8x0 invariably involves the complete process image, i.e. a request by the master leads to the Bus Coupler/Fieldbus Box receiving the complete set of output data and the current process input data being sent to the master in response. Data communication

- between the Bus Coupler and the individual Bus Terminals is via the K-Bus;
- between the Coupler Box and the individual extension modules is via IP-Link.

In the default setting of the Bus Coupler/Fieldbus Box, access to the I/O signals of the terminals/extension modules is asynchronous (*free running* mode). The KS2000 configuration software can be used to switch the mode to *synchronous*. In synchronous mode, access by the Bus Coupler/Fieldbus Box to the Bus Terminals/ extension modules is synchronous with access by the controller to the Bus Coupler/Fieldbus Box.

The data packets are transferred in a fixed format as a binary string. The data frame is set to 8 data bits, with even parity and 1 stop bit (8E1). The baud rate is preset to 38400 baud.

The station addresses are set to Bus Coupler/Fieldbus Box via two rotary switches. If address 0 is set, the BK8x00/IPxxxx-B8x0 operates as master, and the slave Bus Coupler/Fieldbus Box must be allocated address 1.

Request

The master sends a request to the slave with the process output data to be transferred. The slave responds by sending its status and its process input data.

Byte	Description	Valid range of values
0	Start ID	"P" (0 x 50 hex)
1	Number of process data output words	0..255
2	Message Ident	0..255
3	Multipoint address	0..99 for BK8x00 0..69 for IP/ILxxxx-B8x0
4 + 2 x n n = 0...125	Process data output: Low Byte	0..255
5 + 2 x n	Process data output: High Byte	0..255
6 + 2 x n +1	Checksum	0..255

Start ID

The start ID consists of one byte. It identifies the start of a data packet.

Number of process data output words

The number of process data output words specifies the size of the output process image of the addressed Bus Coupler/Fieldbus Box in words. If the number of process image bytes is odd it must be rounded up. Enter zero if only the process input data of the Bus Coupler/Fieldbus Box are to be read.

Message Ident

The message Ident is a (freely selectable) value returned by the receiver in the response string, so that the sender can correlate received and sent strings.

Multipoint address

The multipoint address specifies the receiver. The address value must be unequal 0, since 0 is the master address. For Fieldbus Boxes please note that addresses greater than 69 are not permitted.

Process data output

The process data outputs are entered as data words in Intel format.

Checksum

The checksum is formed by adding up the contents of the individual bytes (total request string, without checksum byte). Any overflow is not taken into account.

Response

The Bus Coupler/Fieldbus Box responds to a request by the master.

Byte	Description	Range of values
0	Start ID	"p" (0 x 70 hex)
1	Number of process data output words	0..255
2	Message Ident	0..255
3	Multipoint address	0..99 for BK8x00 0..69 for IP/ILxxxx-B8x0
4	Status byte [▶ 78]	0..255
5 + 2 x n n = 0...125	Process data output Low byte	0..255
6 + 2 x n	Process data output High byte	0..255
7 + 2 x n +1	Checksum	0..255

Start ID

The start ID consists of one byte. It identifies the start of a data packet.

Number of process data output words

The number of process data output words specifies the size of the output process image of the addressed Bus Coupler/Fieldbus Box in words. If the number of process image bytes is odd it must be rounded up. Enter zero if only the process input data of the Bus Coupler/Fieldbus Box are to be read.

Message Ident

The message Ident is a (freely selectable) value returned by the receiver in the response string, so that the sender can correlate received and sent strings.

Multipoint address

The multipoint address specifies the receiver. The address value must be unequal 0, since 0 is the master address. For Fieldbus Boxes please note that addresses greater than 69 are not permitted.

Status byte (SB)

Bit	Meaning	
SB.0	1_{bin}	Error in the data communication with the Bus Terminals/extension modules.
SB.1	1_{bin}	Configuration error
SB.2	-	reserved
SB.3	-	reserved
SB.4	1_{bin}	Invalid process data output length: The number of process output words received is unequal the physically present data length.
SB.5	-	reserved
SB.6	-	reserved
SB.7	-	reserved

Process data output

The process data inputs are entered as data words in Intel format.

Checksum

The checksum is formed by adding up the contents of the individual bytes (total request string, without checksum byte). Any overflow is not taken into account.

6.1.3.2   KS8000 Communication Software

Fig. 60: Communication library icon

The Beckhoff *KS8000* Communication Library provides functions with which it is possible to communicate with the serial Bus Couplers (BK8000, BK8100, IP/ILxxxx-B8x0) easily via a serial PC interface. The KS8000 can be used as an OCX with all programming languages that use the Component Object Model (COM) specifications from Microsoft (VC++, Visual Basic from version 4.0, Delphi, Java, etc). The KS8000 library also has a DLL interface for any other C/C++ programs.

Accessing the process image

Via the serial PC interface the KS8000 enables access to the input and output process image of the BK8x00 Bus Couplers or the IP/ILxxxx-B8x0 Fieldbus Boxes.

With each serial PC interface, communication with
- a BK8100 coupler (RS232) or
- a Fieldbus Box IP/ILxxxx-B810 (RS232) or
- up to 99 BK8000 couplers (RS485) or
- up to 69 Fieldbus Boxes IP/ILxxxx-B800 (RS485)
can be established. Such communication transfers the whole of the input and output process image. The communication duration therefore depends on the size of the process image. For example, with an RS232 coupler at 38400 baud and

- a process image of one word, the communication duration is approx. 6 ms;
- a process image of 15 words, the communication duration is approx. 20 ms.

Interface for LabVIEW applications

The KS8000 contains an interface for the LabVIEW programming software from National Instruments. LabVIEW solutions consist of what are known as front panels, which serve as the man/machine interface, and of a block diagram, which embodies the actual control program. KS8000 provides LabVIEW users with a greatly simplified method of access to all the process data.

Operation of KS8000 or TwinCAT via the RS485COM port requires a PC card that automatically switches between send and receive mode without echo (e.g. C9900-A440: RS 485 high speed with 2 serial interfaces, optically decoupled, overvoltage protection, ISA bus).

6.2   Introduction to the system

6.2.1   Overview of the physical structure of RS232

The serial, asynchronous RS232C or V.24 communication interface is a widely used as a simple physical connection for exchanging data between two systems. The RS232 interface only describes the physical structure and not the protocol or the application layer.

The data are applied sequentially for the duration of one bit. Synchronization is based on start and stop bits. Asynchronous data transfer requires no clock line. This means that a minimum of 3 lines are required, i.e. one for sending, one for receiving, and one for GND. Since in each case one line is available for sending and one for receiving, the technique is referred to as full duplex transfer. All devices can send or receive as required, since they have a dedicated line available.

The RS232C or V.24 interface operates bipolar with +12 V / -12 V voltage level (+3 V...+12 V = "0" and -3 V...-12 V = "1"). The same baud rate, number of data bits, number of stop bits, and data verification of the so-called parity bit have to be used on both sides. This is usually described as 9600, 8, e, 1 (i.e. 9600 baud, 8 data bits, even parity verification, one stop bit).

6.2.2   Topology of the physical RS232 structure

RS232 is a peer-to-peer connection. The means that only two devices can be connected at any time. The maximum cable length is 15 meters.

7   Error handling and diagnosis

7.1   Diagnostics

State of the K-bus

An internal bus or Bus Terminal error is indicated in the K-Bus state. A more precise fault description can be obtained via a function block (in preparation). To this end, link the "K-Bus state" variable with your PLC program.
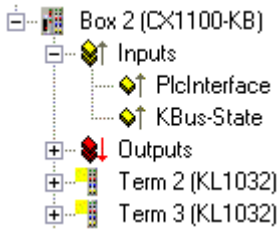
```
⊟─■ Box 2 (CX1100-KB)
   ⊟─◆↑ Inputs
      ├─◆↑ PlcInterface
      └─◆↑ KBus-State
   ⊞─◆↓ Outputs
   ⊞─□ Term 2 (KL1032)
   ⊞─□ Term 3 (KL1032)
```

Fig. 61: State of the K-bus

Error bit	Description	Error type
0	No error	No ERROR.
Bit 0	K-bus error	ERROR
Bit 2	K-Bus is re-triggered	NOTE

Reading K-bus state by ADS

In default or TwinCAT configuration the fieldbus state can be read via ADSREAD.

Parameter ADSREAD function block	Description
NetID	local – empty string
Port	1
IndexGroup	16#0006
IndexOffset	16#000C_9000
LEN	1

7.2   Diagnostic LEDs

The Bus Coupler features status indicator LEDs. The row of LEDs on the left describes the status of the fieldbus and of the PLC. The row of LEDs on the right indicates the supply voltage and the K-Bus state.
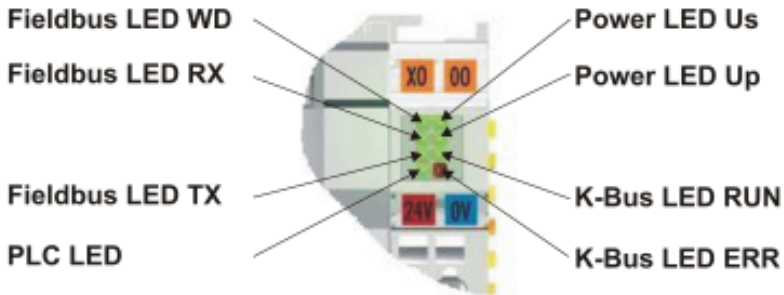
Fig. 62: LEDs

LEDs for power supply diagnostics

LED (Power LEDs)	Meaning
LED Us	LED off: No power supply (24 V_{DC}) for electronics and K-Bus supply connected
LED Up	LED off: No power supply 24 V_{DC} connected at the power contacts

LEDs for K-Bus diagnostics

LED (Power LEDs)	Meaning
LED RUN	LED off: no K-Bus update, LED on, flashing: K-bus running
LED ERR	LED off: no error, LED flashing: see K-Bus error code

K-Bus error code diagnosis

Error code	Error argument	Description	Remedy
0	-	EMC problems	• Check power supply for undervoltage or overvoltage peaks • Implement EMC measures • If a K-bus error is present, it can be localized by a restart of the coupler (by switching it off and then on again)
1	0	EEPROM checksum error	Enter factory settings with the KS2000 configuration software
	1	Code buffer overflow	Insert fewer Bus Terminals. Too many entries in the table for the programmed configuration
	2	Unknown data type	Software update required for the Bus Coupler
2	-	Reserve	-
3	0	K-bus command error	• No Bus Terminal inserted • One of the Bus Terminals is defective; halve the number of Bus Terminals attached and check whether the error is still present with the remaining Bus Terminals. Repeat until the defective Bus Terminal is located.
4	0	K-bus data error, break behind the Bus Coupler	Check whether the n+1 Bus Terminal is correctly connected; replace if necessary.
	n	Break behind Bus Terminal n	Check whether the KL9010 Bus End Terminal is connected
5	n	K-bus error in register communication with Bus Terminal n	Exchange the nth Bus Terminal
6	0	Error at initialization	Exchange Bus Coupler
	1	Internal data error	Perform a hardware reset on the Bus Coupler (switch off and on again)
	2	DIP switch changed after a software reset	Perform a hardware reset on the Bus Coupler (switch off and on again)
7	0	Note: cycle time was exceeded	Warning: the set cycle time was exceeded. This indication (flashing LEDs) can only be cleared by booting the Bus Coupler again. Remedy: increase the cycle time
9	0	Checksum error in Flash program	Re-transfer the program to the Bus Terminal Controller
	1	Incorrect or faulty library implemented	Remove the faulty library
10	n	Bus Terminal n is not consistent with the configuration that existed when the boot project was created	Check the nth Bus Terminal. The boot project must be deleted if the insertion of an nth Bus Terminal is intentional.
14	n	nth Bus Terminal has the wrong format	Start the Bus Coupler again, and if the error occurs again then exchange the Bus Terminal
15	n	Number of Bus Terminals is no longer correct	Start the Bus Coupler again. If the error occurs again, restore the manufacturers setting using the KS2000 configuration software
16	n	Length of the K-bus data is no longer correct	Start the Bus Coupler again. If the error occurs again, restore the manufacturers setting using the KS2000 configuration software

LED bus - fieldbus diagnostics

LED	Meaning
LED WD	not implemented
LED RX	Flashes when data are being received
LED TX	Flashes when data are being sent

LED PLC - PLC diagnostics

LED	Meaning
PLC LED	LED on: PLC running, LED off: PLC stopped

8 Appendix

8.1 First steps with the BC8150

in preparation

8.2 Switching between controllers

Switching from BCxx00 to BCxx50/BCxx20

File names

In the Bus Terminal controllers of the BCxx50 and BCxx20 series, libraries have the extension *.lbx, programs have the extension *.prx.

Flag variables

The allocated flag variables

- of the BCxx00 are assigned %MB0...%MB511 (except BC9000/BC9100: %MB0...%B4095).
- of the BCxx20 are assigned %MB0...%MB4095
- of the BCxx50 are assigned %MB0...%MB4095

Status information such as K-bus/fieldbus status and cycle tick is not copied to the BCxx50/BCxx20. This information is available in TcSystemBCxx50.lbx as a function for the BCxx50/BCxx20.

The allocated flags do **not** act as retain variables.

Retain data

The retain data have to be declared as VAR_RETAIN. Up to 2 kB are available.

PLC Variables

In the Default-Config the PLC variables start from %IB1000 and %QB1000.

Large model

Not applicable for BCxx50 and BCxx20.

Max. memory:

- BCxx50: 48 kB
- BCxx20: 128 kB

Task time

The task time is specified in the PLC Control. It should be set to a realistic value (measuring of PLC cycle time and K-Bus). The background time is not used.

Task configuration

A maximum of one task is available. This task must be configured.

PLC and fieldbus terminals

For the standard Bus Terminal Controllers (BCxx00) it was possible to select whether a Bus Terminal is assigned to the fieldbus or the local PLC.

In the default configuration of the BCxx50/BCxx20 all Bus Terminals are assigned to the local PLC. An assignment to the fieldbus is not possible in this case.

Switching from BCxx00 to BXxx00

File names

In the Bus Terminal controllers of the BCxx00, libraries have the extension *.lbx, programs have the extension *.prx.

Flag variables

The allocated flag variables

- of the BCxx00 are assigned %MB0...%MB511 (except BC9000/BC9100: %MB0...%B4095).
- of the BXxx00 are assigned %MB0...%MB4095

Status information such as K-bus/fieldbus status and cycle tick is not copied to the BXxx20. This information is available in TcSystemBCxx00.lbx as a function for the BXxx50.

The allocated flags do **not** act as retain variables.

Retain data

The retain data have to be declared as VAR_RETAIN. Up to 2 kB are available.

PLC Variables

In the Default-Config the PLC variables start from %IB1000 and %QB1000.

Large model

Not applicable for BXxx00. Max. memory 256 kB.

Task time

The task time is specified in the PLC Control. It should be set to a realistic value (measuring of PLC cycle time and K-Bus). The background time is not used.

Task configuration

A maximum of one task is available. This task must be configured.

PLC and fieldbus terminals

For the standard Bus Terminal Controllers (BCxx00) it was possible to select whether a Bus Terminal is assigned to the fieldbus or the local PLC.

In the default configuration of the BXxx00 all Bus Terminals are assigned to the local PLC. An assignment to the fieldbus is not possible in this case.

Switching from PC to BCxx50/BCxx20/BXxx00

File names

In the Bus Terminal controllers of the BCxx50/BCxx20 and BXxx00 series, libraries have the extension *.lbx, programs have the extension *.prx.

Allocated variables

For the Bus Terminal controllers of the BCxx50/BCxx20 and BXxx00 series, a limited number of allocated data are available:

- Inputs 2 kB, %IB0..2048
- Outputs 2 kB, %QB0..2048
- Flags 4 kB, %MB0..4095

Task configuration

A maximum of one task is available. A sensible task time should be selected. Adjust the task time to your application by measuring the required system time (PLC + K-Bus + fieldbus + other).

Retain data

For the Bus Terminal controllers of the BCxx50, BCxx20 and BXxx00 series, up to 2 kB of retain data are available. Ensure that no (or only very few) retain data are used in function blocks (see RETAIN data).

8.3 Example programs for serial communication

KS8000 protocol

- Example program for BC8150 communication with VB6 without KS8000 OCX or DLL. Source code and EXE file

  (https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/4059887115.zip)

- Example program for BC8150 communication with VB6 with KS8000 OCX. Source code and EXE file:

  (https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/4059889291.zip)

The ADS protocol

- Example program for BC8150 communication with VB6 with ADS OCX. Source code and EXE file
  For this example the AMS Net ID must be set to "1.1.1.1.1.5" (see serial ADS [▶ 36]).

  (https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/4059891467.zip)

Modbus RTU protocol

Example: BC8150 connected to CIMREX 12 panel from Beijers

The serial interface of the BC8150 can also be used as a Modbus slave. In this example, a panel from the company Beijers is connected. Further information about the panel can be found at www.beijerelectronics.de.

Fig. 63: Connection example BC8150 to a CIMREX 12 panel from Beijers

Necessary components

1 x BC8150
1 x Cimrex 12
any Bus Terminals (any, since no Bus Terminals are used in the example)

RS 232 cable

BC8150 COM 1 / RS 232	Cimrex 12 RS 232
3	3
2	2
5	5

Getting started

Download the example into the BC8150, with an address set between 1 and 39 on the BC8150. The program is loaded to the Bus Terminal controller via serial ADS. Start the PLC program, log off, and remove the programming cable. Load the CIMREX 12 project into the panel and remove the programming cable here too. Now connect the Cimrex 12 with the BC8150 using the cable described above. Finally, set slave address 41 on the BC8150. After 10-20 seconds a numerical value should start incrementing on the panel and stop when F2 is pressed on the Cimrex 12.

sample program in ST for BC8150 (any terminal configuration) (https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/4059893643.zip)

Example Cimrex 12 panel (https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/4059895819.zip)

No libraries are required for the example
- baud rate 9600,n,8,1 D
- address 41 on the BC8150
- Cimrex 12

The ADS protocol

Example: BC8150 connected to CIMREX 12 panel from Beijers

RS 232 cable

BC8150 COM 1 / RS 232	Cimrex 12 RS 232
3	3
2	2
5	5

sample program in ST for BC8150 (any terminal configuration) (https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/4059893643.zip)

Example Cimrex 12 panel (https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/4059897995.zip)

8.4   Firmware Update

Firmware update program

The firmware update program is required for loading a new firmware to the Bus Coupler. The program is transferred via the serial interface.

Note for BX3100:
Updates are not available with BX3100 firmware 0.64 (or lower). If these devices need updating, send the BX3100 to the manufacturer with a corresponding note.

Beckhoff Automation GmbH & Co. KG
Service Department
Stahlstr. 31
33415 Verl, Germany

Firmware update program 241 (https://infosys.beckhoff.com/content/1033/bc8x50/Resources/zip/3238791819.zip) 71 kbytes (for Windows NT4.0 SP6, 2000, XP).

The program *FirmwareUpdate.exe* and the file *TcRouterHelper.dll* have to be in the same directory. Open the program by double-clicking on *FirmwareUpdate.exe*.

Update for Bus Terminal Controllers

BX series

Select the appropriate device of - in this example "Serial interface (BX)".
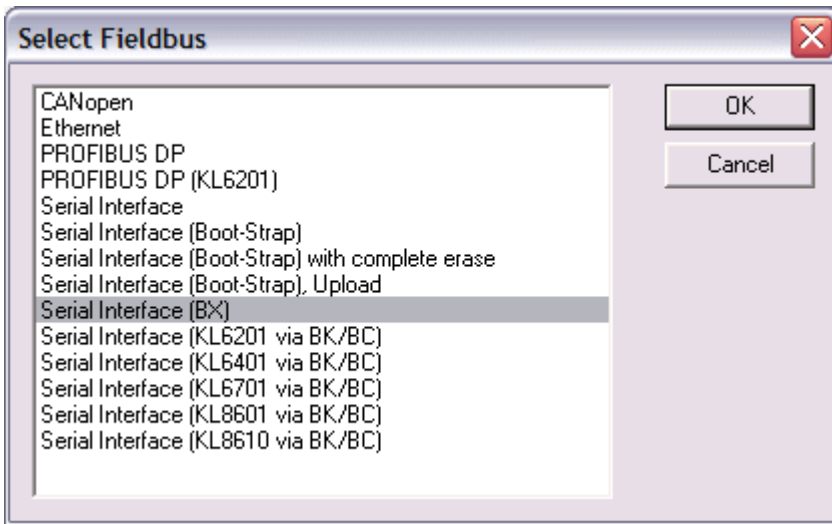
Fig. 64: Selecting a BX series Bus Terminal Controller

BCxx50 series

Select the corresponding device, in this case "Serial Interface".
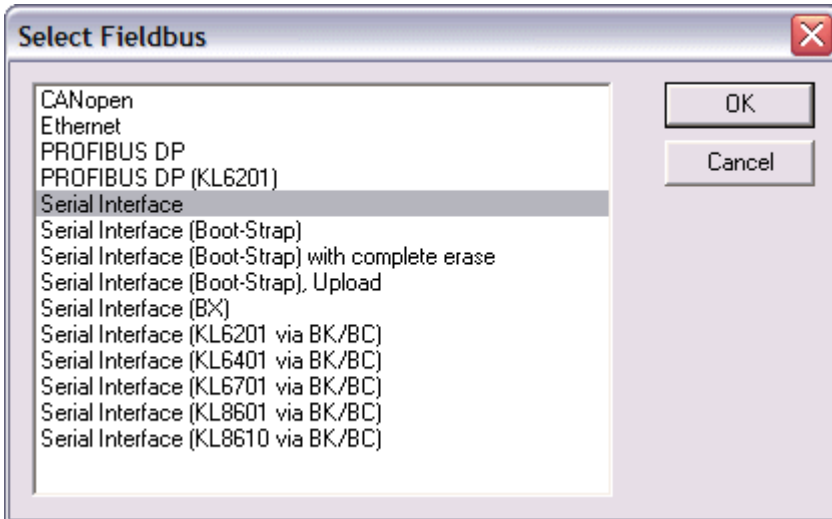
Fig. 65: Selecting a BC series Bus Terminal Controller

BX and BCxx50 series

Then select the COM port.
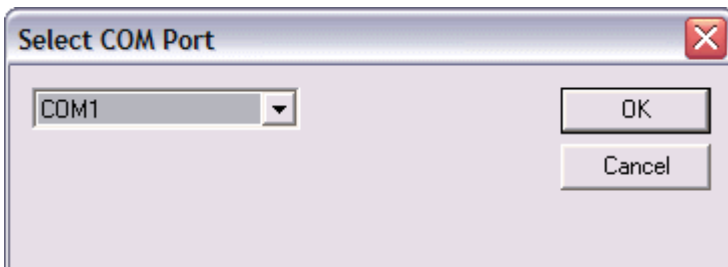
Fig. 66: Select the COM port

Open the file you wish to download.

BECKHOFF
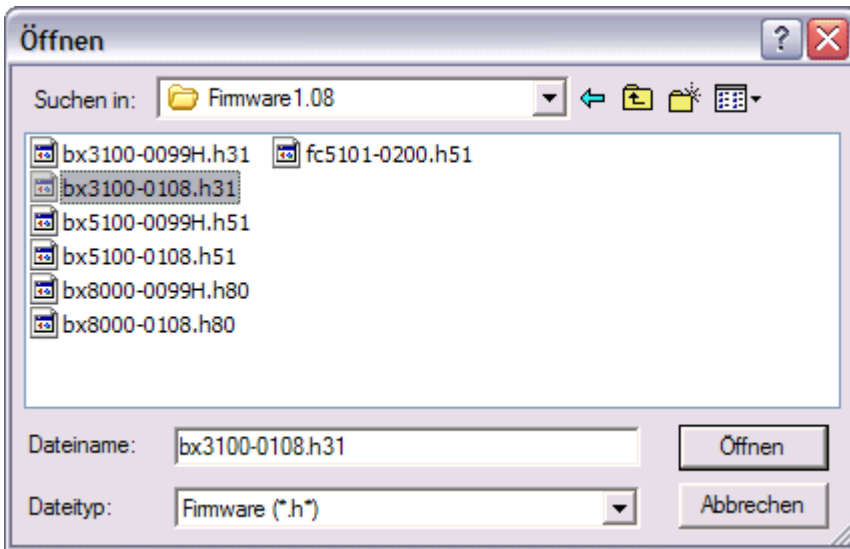
Fig. 67: Open the firmware file

Start the download via the green 'traffic light'. The download begins after about a minute, and is then also shown on the BX's display. After successful download (approx. 2 to 3 minutes) the Bus Terminal Controller reboots automatically.
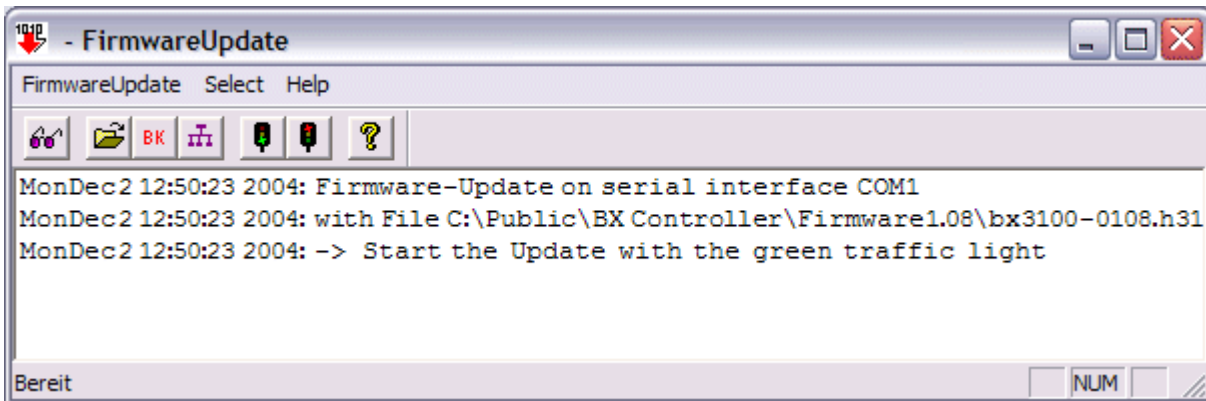
Fig. 68: Status messages relating to the firmware update

8.5   General operating conditions

The following conditions must be met in order to ensure flawless operation of the fieldbus components.

Environmental conditions

Operation

The components may not be used without additional protection in the following locations:

- in difficult environments, such as where there are corrosive vapors or gases, or high dust levels
- in the presence of high levels of ionizing radiation

Condition	Permissible range
Permissible ambient temperature during operation	see technical data
Installation position	variable
Vibration resistance	conforms to EN 60068-2-6
Shock resistance	conforms to EN 60068-2-27
EMC immunity	conforms to EN 61000-6-2
Emission	conforms to EN 61000-6-4

Transport and storage

Condition	Permissible range
Permissible ambient temperature during storage	-25°C... +85°C
Relative humidity	95 %, no condensation
Free fall	up to 1 m in the original packaging

Protection classes and types

Condition	Permissible range
Protection class in accordance with IEC 536 (VDE 0106, Part 1)	A protective conductor connection to the profile rail is necessary!
Protection class conforms to IEC 529	IP20 (protection against contact with a standard test finger)
Protection against foreign objects	Less than 12 mm in diameter
Protection against water	no protection

Component identification

Every supplied component includes an adhesive label providing information about the product's approvals. For sample, on the BK2000 Bus Coupler:
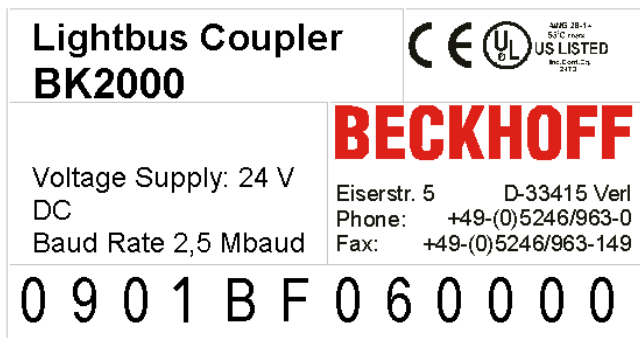
The following information is printed on the label:

Printed item	Meaning for this label
Precise product identification	Lightbus Coupler BK2000
Supply voltage Us	24 V_{DC} (Use a 4 A fuse or a Class 2 power supply to meet UL requirements)
Data transfer rate	2.5 Mbaud
Manufacturer	Beckhoff Automation GmbH & Co. KG
CE mark	Conformity mark
UL mark c **UL** US LISTED Ind. Cont. Eq.: 24TB Use 4 Amp. fuse or Class 2 power supply. See instructions.	Mark for UL approval. UL stands for the Underwriters Laboratories Inc., the leading certification organization for North America, based in the USA. C = Canada, US = USA, UL file number: E172151
Production identification	From left to right, this sequence of characters indicates the week of production (2 characters), the year of production (2 characters), the software version (2 characters) and hardware version (2 characters), along with any special indications (4 characters). In this case the device is a BK2000 - produced in calendar week 9 - of 2001 - with firmware version BF - in hardware version 6 - without special designation

8.6   Test standards for device testing

EMC

EMC immunity

EN 61000-6-2

Electromagnetic emission

EN 61000-6-4

Vibration / shock resistance

Vibration resistance

EN 60068-2-6

Shock resistance

EN 60068-2-27

8.7   Bibliography

in preparation

8.8   List of Abbreviations

in preparation

8.9 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on her internet pages: https://www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline:	+49 5246 963 157
Fax:	+49 5246 963 9157
e-mail:	support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline:	+49 5246 963 460
Fax:	+49 5246 963 479
e-mail:	service@beckhoff.com

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone:	+49 5246 963 0
Fax:	+49 5246 963 198
e-mail:	info@beckhoff.com
web:	https://www.beckhoff.com

Table of figures

BECKHOFF

More Information:
www.beckhoff.com/BCxxxx